

AFRL-RI-RS-TR-2007-224
Final Technical Report
October 2007



SYNTHESIS OF BIOFLUIDIC MICROSYSTEMS (SYNBIOSYS)

Carnegie Mellon University

Sponsored by
Defense Advanced Research Projects Agency
DARPA Order No. K900/08

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

STINFO COPY

**The views and conclusions contained in this document are those of the authors
and should not be interpreted as necessarily representing the official policies,
either expressed or implied, of the Defense Advanced Research Projects
Agency or the U.S. Government.**

**AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK**

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report was cleared for public release by the Air Force Research Laboratory Public Affairs Office and is available to the general public, including foreign nationals. Copies may be obtained from the Defense Technical Information Center (DTIC) (<http://www.dtic.mil>).

AFRL-RI-RS-TR-2007-224 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE DIRECTOR:

/s/

/s/

PETER J. ROCCI, Jr.
Work Unit Manager

JOSEPH CAMERA, Chief
Information & Intelligence Exploitation Division
Information Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

Abstract

Biofluidic microsystems are essentially a miniaturized microchip implementation of an analytical chemistry laboratory. Such microsystems are fast, accurate, automatable and inexpensive to fabricate and have spanned sample collection, preparation and analysis for biological/chemical agent detection and on-chip drug or chemical synthesis and delivery for warfighter defense.

The design of such biofluidic microsystems has posed a significant challenge. An initial generation of design tools based on continuum simulation and reduced order modelling was developed through DARPA support in the late '90s. These tools were useful for *component* design and optimization, however, they were too cumbersome to be applied at the *system* level. As designers evolved from designing individual channels, pumps, valves and mixers, to whole assays on a chip, this shortcoming became critical.

This report describes approaches to system-level design tools and methodologies. It incorporates significant advances in a behavioral modeling and simulation methodology for Lab-on-a-Chip (LoC) design. This modeling and simulation methodology was developed with optimization for design synthesis in mind, enabling rapid automated design of highly complex biofluidic microsystems.

The behavioral modeling methodology involves decomposing a complex LoC system into a small set of elements. Each of the elements is associated with a parameterized behavioral model that describes its electric and biofluidic behavior. Key issues addressed include schematic representation, behavioral multi-physics modeling and numerical and experimental validation. The modeling effort focused on sample transport in LoC devices. Turn and Joule heating induced dispersion in electrophoretic separation chips are modeled using the method of moments. The skew of the species band is effectively represented by a set of Fourier cosine series coefficients that are obtained analytically. These coefficients capture the effect of band skew on separation performance in various complex chip geometries (including multiple turns). Variations of sample concentration profiles in laminar diffusion-based micromixers are also derived using the Fourier cosine series representation. The model holds for arbitrary sample flow ratios and inlet concentration profiles, and accurately considers the overall effects of device topology, size and electric field on mixing performance. In addition, a simplified reaction model is developed and integrated with the separation and mixing models to perform system-level schematic simulations of an integrated LoC system. Simulation results at both element and system levels are validated against numerical and experimental data.

Excellent accuracy (generally less than $< 5\%$ in relative error) and tremendous speedup ($> 100\times$) have been achieved when compared with finite element analysis. The resulting modeling and simulation framework is a significant contribution to balancing the needs for efficiency and accuracy thus enabling iterative design of complex biofluidic LoC systems.

These models were integrated in a simulation backplane that captured the complex physio-chemical phenomena and enabled the development of design tools that simultaneously considered these phenomena as well as challenging chip layout and channel interconnectivity issues. The chemistry that takes place during chip operation, as well as the chip layout and manufacturing process must be understood so that the appropriate design trade-offs and constraints are considered. Channel geometry, and the system's channel topology are shown to contribute a great deal to the overall performance of the final LoC design. These issues result in a design problem that is highly nonlinear and highly combinatorial.

Table of Contents

| | |
|--|-----|
| 1. Summary | 4 |
| 2. Introduction | 5 |
| 3. Methods, Assumptions, Procedures | 9 |
| 4. Results, Discussion | 20 |
| 4.1. Component Models Development and Verification | 20 |
| 4.2. System Simulation | 63 |
| 4.3. Component Optimization | 78 |
| 4.4. Place and Route | 86 |
| 5. Conclusions | 105 |
| 6. Recommendations | 105 |
| 7. References | 106 |

List of Figures and Tables

Figures

- FIGURE 1. (a) BioMEMS microneedles for transdermal drug delivery [2], (b) Multiplex Lab-on-a-Chip DNA sequencer integrates 384 independently operatable capillary electrophoretic separation channels for high throughput analysis [3], (c) Gene expression microarray with > 1 million probes with zoom in of individual probe locations after hybridization reaction [4]. 5
- FIGURE 2. (a) Cross section of EWOD device [12]; (b) Droplet injection from reservoir (left) and split (right); (c) Independent and simultaneous droplet transport; and (d) two droplets are merged (left and center), and then merged with reservoir (right) [14]. 7
- FIGURE 3. (a) Spiral Capillary Electrophoresis separation assay for amino acids [20]; (b) Microreactors for drug discovery [21]; (c) Multifunction immunoassay that integrates mixing, reaction, and separation [22]. 8
- FIGURE 4. System-level Immunoassay schematic showing straight channels, U turns, elbow turns, and other elements used to integrate mixing, reaction, injection and separation functions on a single LoC [41]. 10
- FIGURE 5. Related modeling methods. 13
- FIGURE 6. (a) The activation, z_k , of a node is a function of the weighted inputs of the previous layer's activations. (b) General feedforward neural network topology utilizing multiple hidden layers with an unspecified number of nodes with non-linear activation functions, and a single output node using a linear activation function. 14
- FIGURE 7. k-fold cross validation algorithm. First divide the N points in the dataset into k subsets (3 to 5 subsets is typical). Then, identify one subset (i) as the validation set, used to compare with the trained neural network. The other $k-1$ subsets ($\sim i$), are used to train the neural network. Each subset is the validation set exactly once. The error for each iteration is averaged at the end to provide an estimate of the generalization error for the network. This is an efficient use of data, since all data is used for training and validation. There is no exclusive validation set. 16
- FIGURE 8. (a) Uniform sampling on a grid in two-dimensions with a total of 16 data points resulting in effectively 4 data points per dimension. (b) Uniform sampling off-grid in two dimensions, again with 16 data points, resulting in effectively 16 points per dimension, a better more denser sampling per dimension. 17
- FIGURE 9. A neural network is trained on the function shown on the left. The network has a single hidden layer of tansig neurons and one linear output layer. It is trained for 450 max iterations, and the sample size is varied. The error measured is the mean squared error of the network to the training data (not a measure of the generalization error). The Sobol quasirandom sequence is the best performing, followed by the Gaussian sequence, and finally the uniform grid. The Gaussian sequence is a simpler algorithm and is compatible with domain shaping methods. 19
- FIGURE 10. (a) Behavioral model structure for the converging intersection in the micromixer. Index l, r and out represent the quantities at the left inlet, right inlet and outlet. Both electric (V) and biofluidic (dm) pins are defined at the terminals of the model. Electrically, it is modeled as a combination of three resistors (R_l , R_r and R_{out}) with zero resistance. Different sample concentration profiles, $cl(h)$ and $cr(h)$, at inlets are merged and compressed at the outlet $c_{out}(h)$. (b) Behavioral model structure for the diverging intersection in the micromixer. Similarly, index l, r and in represent the quantities at the left outlet, right outlet and inlet. Sample concentration profile at inlet, $c_{in}(h)$ is split and stretched out into two parts, $cl(h)$ and $cr(h)$, flowing out of the outlets. 22
- FIGURE 11. Behavioral model structure for separation channels in electrophoretic separation microchips. Index in and out represents the quantities at the inlet and outlet. Both electric (V) and biofluidic (t, S_n , s_2 and A) pins are defined at the terminals of the model. Electrically, the channel is modeled as a resistor (R). The variations of biofluidic pin values due to dispersion effects are captured by the model. 25

- FIGURE 12. Verilog-A description for a 180° turn involving clockwise flow of the species band. It determines the signs used in Eq. (24), as well as the canceling and strengthening effects of the skew. 28
- FIGURE 13. (a) An electrokinetic focusing micromixer. Sample a, flowing from the top input channel to the intersection, is pinched by sample b (or buffer) from both side channels. Then samples mix in the bottom mixing channel. (b) Its hierarchical schematic representation. The triple input and one output cross intersection is modeled as a cascade connection of two converging intersections. 30
- FIGURE 14. (a) Schematic simulation results (lines) compared with numerical data (symbols) on concentration profiles c (sample a) along the normalized channel width h for the electrokinetic focusing and T-type mixers. In contrast to the T-type mixer, the focusing mixer considerably improves sample homogeneity due to the reduced diffusion distance between samples; (b) Schematic simulation results on variation of mixing residual Q along axial channel length (data points are connected by lines to guide the eye) for the electrokinetic focusing mixer involving different stream width s . A smaller stream width (e.g.,) yields a lower initial mixing residual and a more uniform concentration profile at the end of the mixing channel. 30
- FIGURE 15. An electrokinetic serial mixing network [74] and its hierarchical schematic representation. The network consists of reservoirs, mixing channels, T- and cross-intersections. Sample and buffer is released and collected by the reservoirs. In the composable approach, the serial mixing network is represented as a collection of interconnected mixing elements composed of microchannels, converging intersections and diverging intersections. 32
- FIGURE 16. Comparison of experimental data [79] with DC schematic simulation on variance s^2 vs. separation time t of species a in a serpentine electrophoretic separation microchip of two complementary turns. The grey bars represent the residence time of the sample within the turns. The first turn skews the species band and accordingly incurs abrupt increase in variance. The transverse diffusion in the inter-turn straight channel smears out most of the skew and presents a nearly uniform band before the second turn (see the inset of numerical simulation plot). The second turn then distorts the band again in the opposite direction, leading to another turn-induced variance. 34
- FIGURE 17. Transient analysis simulates the electropherograms output from three detectors, which are respectively arranged before the first turn (top trace), in the middle of the inter-turn straight channel (middle trace) and after the second turn (bottom trace). Attributed to the difference in electrokinetic mobility, the spacing between concentration peaks of species a and b increases as they migrate through channels. The dispersion effect leads to the continuous decreases in the band amplitude. 34
- FIGURE 18. (a) A spiral electrophoretic separation microchip [83]. It consists of five turns with continuously decreased radius (1.9, 1.8, 1.7, 1.6 and 0.8 cm). Within them, species Dichlorofluorescein flows in the same direction (clockwise). (b) Its hierarchical schematic representation. 35
- FIGURE 19. Comparison of schematic simulation results to the experimental data on theoretical plate number N_s vs. electric field E . Right axis shows the relative error between simulation and experiments. The linear growth of the plate number N_s with electric field E implies that molecular diffusion is the major dispersion source in such a system. 36
- FIGURE 20. (a) A hybrid electrophoretic separation microchip. It consists of both spiral and serpentine channels. Species flows in the clockwise direction in both turns T1 and T2 (spiral topology), thereby T2 strengthens the sharp skew generated by T1. The skew almost persists through the inter-turn straight channel between T2 and T3 and is significantly cancelled out by T3 (serpentine topology). (b) Its hierarchical schematic representation. 36
- FIGURE 21. Comparison of numerical data with DC schematic simulation on variance vs. separation time in the hybrid electrophoretic separation microchip. Very sharp skew (see Fig. 20) is generated and the variance accumulates after turns T1 and T2 due to their spiral topology. The skew almost persists through the inter-turn straight channel between T2 and T3 and is significantly cancelled out by T3 attributed to their serpentine topology, which as a result yields a drastic variance drop after T3. 37
- FIGURE 22. Basic operating stages for various injector topologies. The most common topologies are the cross, double-tee, and gated cross. The earliest on-chip injectors were the tee. 38

FIGURE 23. Two species are combined and a band of material representative of an injector formed plug is placed at the beginning of the channel. Species one and two have a diffusivity of $1 \times 10^{-10} \text{ m}^2/\text{s}$. Species one has a mobility of $5.25 \times 10^{-7} \text{ m}^2/\text{Vs}$, whereas species two has a mobility of $5.5 \times 10^{-7} \text{ m}^2/\text{Vs}$. After traveling in a 50000 V/m electric field for 1.2 mm , the separation of the two species bands is measured. In the first example (a), the plug has a width of $10 \mu\text{m}$. In the second example (b), the plug has a width of $50 \mu\text{m}$. Even though the plug in (b) contains more material, it is not as easy to resolve as the smaller plug in (a).

39

FIGURE 24. Proper operation of an injector requires the identification of control parameters and their applicable ranges. Here identifying the fields (arrows) is the first step, but then the magnitude and ratios must be additionally constrained for proper operation ($E1L > 0$, $E1D/E2D > a$, where $E1D=E3D$)

FIGURE 25. Leakage results for the cross and double-tee injectors. The values of the loading stage parameters are fixed, but different in each of the four plots. The blue region defines infeasible operation, and the red region defines feasible operation for the dispensing stage field ratios, eD and Peclet number PeD .

43

FIGURE 26. Comparison of actual injection plug to its effective Gaussian representation. The actual output of a double-tee injector is in the top channels, the effective Gaussian model is shown in the bottom channels. The band on the right has traveled 4.7 mm , which is when $t = 2$. For these simulation $p1D = 186$, $p2D = 186$, $p3D = 1/8$, and $p4D = 0.57$, $p5D = 2$ (definitions can be found in Table 3).

44

FIGURE 27. Snapshots of the transient injection process for the cross injector using electrokinetic transport. Starting on the left, the analyte in red is drawn from the sample reservoir or upstream mixer at the top (1). After filling the injection chamber (2), the applied potential is changed to inject a plug of analyte while evacuating leftover sample to the north and south to prevent leakage (3,4). The three shorter channels are three channel lengths long and the long channel is 12 channel lengths long.

45

FIGURE 28. (a) Traces through the design space for fixed parameters of $p1c = 0.5$, $p2c = 0.5$, $p3c = 400$, $p4c = 400$. (b) Scatter plots of the design space sampled for training the neural network model.

47

FIGURE 29. (a) Mean squared error results from cross KFCV model selection test. The minimum occurs near a network with a single layer of 50 neurons for both models. As the complexity of the network increases, the generalization error increases even though training error may decrease due to ‘overfitting’ the data. (b) The trend is easier to see with the data logarithmically scaled. Noise in the figures is from the randomness added by initializing the Levenberg-Marquardt training algorithm.

48

FIGURE 30. Plots of band variance and peak height vs. $p3c$ for varying values of $p4c$. The maximum error of both models is approximately 10.5%. Dots are the numerical simulations, solid lines are the NN function results.

49

FIGURE 31. Snapshots of the transient injection process for the double-tee injector using electrokinetic transport. Starting on the left, the analyte in red is drawn from the sample reservoir at the top(1). After filling the injection chamber (2), the applied potential is changed to inject a plug of analyte while evacuating leftover sample to the north and south to prevent leakage (3,4,5).

49

FIGURE 32. Simulations of the double-tee injector for $p1d = p2d = 2.5$, $p3d = p4d = 500$ for (a) $p5d = -2$ (b) $p5d = 0$ (c) $p5d = 2$. The surface plots are not to scale, (a) and (b) are scaled to a maximum concentration of 0.5 and (c) is scaled to a maximum concentration of 1. (a) is a backward offset which produces the smallest plug, (b) is a zero offset, and (c) is a forward offset which produces the largest plug. This effect is apparent when significant pullback is used, $p2d > 1$.

50

FIGURE 33. (a) Trace of the variance, peak height, and mass vs. channel offset for $p1d = p2d = 0.5$, $p3d=400$, $p4d=400$. (b) Trace of the variance, peak height, and mass vs. channel offset for $p1d = p2d = 2.5$, $p3d=400$, $p4d=400$. (c) Projected sampling of the design space.

51

FIGURE 34. Mean squared error results from double-tee KFCV model selection test. The minimum occurs near a network with a single layer of 35 neurons for the variance model and 25 for the concentration model.

52

| | | |
|------------|--|----|
| FIGURE 35. | Plots of band variance and peak height vs. p_{3c} for varying values of p_{4c} . The maximum error for both models is approximately 7.72%. Dots are the numerical simulations, solid lines are the NN function results. | 53 |
| FIGURE 36. | Snapshots of the transient injection process for the gated-cross injector using electrokinetic transport. Starting on the left, the analyte in red is drawn from the sample reservoir at the top and takes a 90° turn forming a gate in the injection chamber(1). The chip operates in this mode until a plug is needed. At this time the electric potential in channel 3 is floated causing the analyte to flood the chamber (2). When the plug of desired size is formed, the potential is reapplied and the gated is reformed injecting the plug into the separation channel (3). | 53 |
| FIGURE 37. | Gated-cross feasibility space (dark is feasible) | 55 |
| FIGURE 38. | Gated-cross leakage tested at Peclet numbers from 19 at contour 1 to at contour 5. The contours are defined by the 7% of the maximum concentration, and agree very well with the numerical and experimental results found in [91] | 55 |
| FIGURE 39. | (a) Mean squared error results from gated-cross KFCV model selection test. The minimum occurs near a network with a single layer of 35 neurons for the variance and concentration models. (b) Response surface of numerical simulation on top and neural network on bottom for variance model. The neural network is able to produce results in a fraction of second, whereas the numerical simulation takes hours. (c) Response surfaces for peak concentration model, numerical simulation on top, neural network on bottom. p_{3G} and p_{4G} are fixed at 232 and 0.1208 respectively. | 56 |
| FIGURE 40. | Schematic illustration of a basic micro mixing and reaction system. | 58 |
| FIGURE 41. | The micro reactor is represented by a PFR network model. The calculation of reaction and convection is conducted in one column of PFRs and the calculation of diffusional mixing is conducted between two columns of PFRs. | 59 |
| FIGURE 42. | Apply the numerical method of lines to calculate the diffusion in the channel width direction. Here, we take 10 discretized concentration points in the channel: c_{i1} - c_{i10} . Points for c_{i0} and c_{i11} are needed to apply the boundary conditions. | 61 |
| FIGURE 43. | (a) The concentration profiles generated by FEM simulations with decreasing size of elements. The red curves represent the results associated with the small element size. They are grid independent and can be used as a basis for comparison. (b) Timing results for the FEM simulations. | 63 |
| FIGURE 44. | (a) Comparison of MOL model simulation results with FEM simulations. Using more discretization in the MOL model leads to a more accurate solution. (b) Timing results for the MOL model simulation as a function of the discretization. | 63 |
| FIGURE 45. | (a) Sketch of a competitive immunoassay microchip consisting of four subsystems (mixing, reaction, injection and separation) [82]. (b) Its system-level schematic. | 65 |
| FIGURE 46. | Behavioral model structure for the electrokinetic reactor. At the inlet, the Fourier coefficients () of widthwise concentration profiles of samples and reagents conveyed from upstream mixers characterize their premixing degree. At the outlet, biofluidic pins quantify the average concentrations () of the reaction products and unreacted samples. | 66 |
| FIGURE 47. | (a) Sketch of the competitive immunoassay reaction between Th, Th* and Ab. (b) Sketch of the competitive immunoassay reaction model. | 67 |
| FIGURE 48. | Pins for the electrokinetic injector. Indices in, out, b, aw refer to the quantities at the terminals respectively linking to the reactor, separation channel, buffer reservoir and sample (analyte)-waste reservoir. | 71 |
| FIGURE 49. | Comparison between the system-level simulation results and experimental data [16] on the area ratio of unreacted Th* and Ab-Th* complex. Abscissa shows the initial concentration of Th before the 50-fold off-chip dilution. Actual concentrations of Th* and Ab are not available. Their values were extracted from the results in Ref. [16]. | 75 |
| FIGURE 50. | Electropherograms of unreacted Th* and Ab-Th* complex from system simulations at three detection | |

spots: 10 mm after injection (top trace), before the 90° elbow (middle trace) and after the 90° elbow (bottom trace). The concentration used in experiments and simulations is 40 mg/L before the 50-fold off-chip dilution (800 mg/L after the dilution). 76

FIGURE 51. Schematic (not to scale) of the improved design of Fig. 45 using the system-level simulation. 77

FIGURE 52. The three necessary connectivity constraints. (a) Two legal turns (1 and 2) and two illegal turns (3 and 4) that can be attached to the endpoint of a straight channel. (b) The outlet flow direction of the straight and the inlet flow direction of the turn must match, thereby eliminating illegal turns. (c) The topology becomes unique when a turn direction is specified. 80

FIGURE 53. Adding a second turn. (a) An illegal clockwise turn (b) Legalize by shortening the initial straight section. (c) Legalize by lengthening the first clockwise turn. 80

FIGURE 54. Bounding-box determination for (a) serpentine topologies and (b) spiral topologies. All points are calculated with respect to P0 (the origin). The X and Y dimensions are the actual maximum topology dimensions plus 2PAD, where PAD is a minimum feature size specification. 81

FIGURE 55. Improvement of a spiral design: (a) A schematic of the design presented in [20] (b) A schematic of an improved design generated using our optimization algorithm. 85

FIGURE 56. Magnification of the optimized spiral design shown in Fig. 55. 85

FIGURE 57. Eight possible subsystem orientations. 87

FIGURE 58. (a) Schematic of a serpentine subsystem showing horizontal X and vertical Y dimensions, and port locations (Sx,Sy), (Bx,By), (Swx,Swy), and (Bwx,Bwy). (b) Three subsystem chip layout showing the horizontal W and vertical H dimensions and subsystem locations (x,y)_i. Fluid I/O wells are shown as circles, and auxiliary channels (black lines) connect wells and subsystems. 89

FIGURE 59. (a) One serpentine unit (t = 1). (b) Two serpentine units (t = 2). (c) Completing the topology with an initial straight section. 92

FIGURE 60. (a) Compacted arrangement of subsystems. (b) Corresponding well placement graph. 94

FIGURE 61. (a) Poor well placement based on rectilinear model (dotted arrow). (b) Compacted arrangement pulled apart by weighted-sum objective function. 95

FIGURE 62. Flowchart for Floorplan Solution Method. 96

FIGURE 63. Flowchart for the routing algorithm. 99

FIGURE 64. Five-subsystem example: (a) Compacted arrangement (b) Final design (routed). 101

FIGURE 65. Ten-subsystem example: (a) Compacted arrangement (b) Final routed design. 101

FIGURE 66. Scaling of NLP compaction subproblem in the floorplanning algorithm per size of design instance. Tests were performed on a standard PC (Intel Pentium 3 CPU: 1GHz CPU, 1Gb RAM). 102

FIGURE 67. Example routing: (a) Minimum length only. (b) Minimum length and number of bends. 103

FIGURE 68. Total routing length vs. number expansions. 104

Tables

| | | |
|-----------|--|----|
| Table 1. | Definition of Biofluidic Pins | 27 |
| Table 2. | Comparison of schematic simulation results (sche) with numerical (num) and experimental (exp) data on sample concentrations in analysis channels of serial and parallel mixing networks [74] | 31 |
| Table 3. | Cross Injector Simulation Parameters | 45 |
| Table 4. | Double-Tee Simulation Parameters | 50 |
| Table 5. | Gated Cross Simulation Parameters | 54 |
| Table 6. | Timing results for the PFR network model. | 62 |
| Table 7. | Molar mass flow rates of samples and reaction products at the inlet and outlet of the reaction model | 70 |
| Table 8. | Biofluidic wiring buses used in the integrated LoC simulations | 73 |
| Table 9. | Comparison between system-level simulation results and experimental data on normalized concentrations of Th* at channels J1-J3 and J3-J4. | 74 |
| Table 10. | Comparison between numerical analysis [29] and system simulations on the area ratio | 75 |
| Table 11. | Comparison of the channel dimension, separation time, variance, peak height and chip area between the original design and the improved design. | 77 |
| Table 12. | Examples of three different constraint classes along with typical numerical values. | 79 |
| Table 13. | Spiral Design Data (extracted from [20]). | 84 |

Acknowledgement

We thank the DARPA and Dr. Anantha Krishnan, the DARPA program manager that has supported this project. We are grateful to Peter Rocci of the Air Force Research Laboratory at Rome, NY, for his guidance as contract manager. We are also grateful to Clare Theim of the Air Force Research Lab for his guidance in this project. We would also like to thank John Gilbert and Manish Deshpande when they were with Coventor for the original proposal discussions that led to the formulation of this project. The faculty investigators thank the many graduate students whose work is presented in this report for a job well done. The Ph.D. students are Bikram Baidya, Xiang He, Ryan Magargle, Anton Pfeiffer, and Yi Wang. The M.S. students are Saurabh Deshpande, Dipti Motiani and Dan Patterson. And the undergraduate students are Rex Carazo-Zapetis, Chris Denault, Jeremy Ellman, Syed Hussain, Rathan Rajanamanickam, and Michael Vahey. Last, but not least, we thank Mary Moore for administrative support for this project.

1. Summary

One of the original DARPA supported biofluidic microsystem researchers, Jed Harrison, of the University of Alberta said that “It will become important to place design capability within the hands of the chemist or biologist who understands the application if we are to see a dynamic leap forward in the acceptance of microfluidic technology” [1]. This has guided project proposal as well as its implementation.

The primary goals of the “Synthesis of Biofluidic Microsystems” project were to develop:

- Shorten the development cycles of biofluidic microsystems; at the time of the proposal quoted times for custom biofluidic chip design was on the order of 1 year. The proposal goal was to cut it down to a few days, or a greater than 10X improvement.
- Enable design of more complex biofluidic microsystems; at the time of the proposal, the state of the art was a single separation system or a single reaction system with a few channels with one or two biomolecular species. The proposal goal was to enable multiple parallel assays with more than 10 biomolecular species and to integrate thousands of channels, or a greater than 100X improvement.
- Develop a lab-on-a-chip design methodology that was primarily focused on synthesis to impact the development of structured iterative design methodologies.

All the goals were met and exceeded during the course of the project. For example, the tools and methodologies developed in this project were used to design a chip with more than 4 meters of microchannels to simultaneously perform multiple assays that ranged from separating DNA molecules and separating amino-acids to an immunoassay that combined both separation and reaction subsystems in less than one hour!

To achieve these innovations we have developed a modeling and simulation methodology that partitions complex lab-on-a-chip systems into commonly used elements. Parameterized models are developed for each of these elements that no longer need artificial parameter fitting from continuum simulation that used to be the state of the art at the time this project started. Also, we have developed mathematical formulations for optimal design of LoC subsystems and multifunction LoCs. A rigorous general disjunctive programming formulation for the optimal design of multiplexed LoCs as well as a more tractable combinatorial formulation was also developed.

These LoC system design tools have a direct insertion path in the recent drive to low-cost sequencing engines needed for further advancement in genomics and proteomics. The DoD shared this interest for use in DNA dogtags for warfighter identification. Additional applications are expected for the design of a variety of point of care devices for hand-held whole blood monitors and bio-weapons detectors.

2. Introduction

Microdevices are finding increasing application in biology and chemistry, and the life sciences. A driving biomedical instrumentation vision involves extracting a fluid sample from the human body, analyzing it to diagnose disease, and then injecting a drug to treat it, all within the same microdevice. Constituent components for this vision have been introduced over the last decade, and can be partitioned into three categories. *BioMEMS* devices are made using planar IC manufacturing techniques, augmented with MEMS-based sacrificial etching processes. BioMEMS devices are typically discrete, such as valves, pumps or needles (Fig. 1(a)). *Lab-on-a-Chip* (LoC) or *Micro Total Analysis System* (μ TAS) devices are biofluidic integrated circuits, with fluid instead of current flows (e.g. DNA separation assay in Fig. 1(b)). A third class of device, called *BioChips* or *Microarrays* (Fig. 1(c)), integrates surface chemistry needed for biomolecule identification (bio-sensing). The discrete BioMEMS devices are not really chips, and will not be considered in this paper. The planar integrated BioChip and Lab-on-a-Chip platforms are both suitable for biosensing, and can use the BioMEMS devices to interface to the human body for sampling fluid from the body and delivering a drug to the body.

Biosensing, or analysis of unknown biological markers in a sample, is a basic lab function. Let us consider a benchtop lab to see what needs to be integrated on a chip. Analysis begins with the sample being *mixed* with known fluorescently labelled probes (using shakers, stirrers or centrifuges). This mixture is then converted into a *reaction* product, typically with the help of catalysts, heat, light, or enzyme. Next, the reaction products are placed in a field (e.g. electric or gravitation) *separating* the constituents according to their charge or mass. Whether the reaction took place or not can thus be identified using the fluorescent marker, and leads to the lab technician labelling the

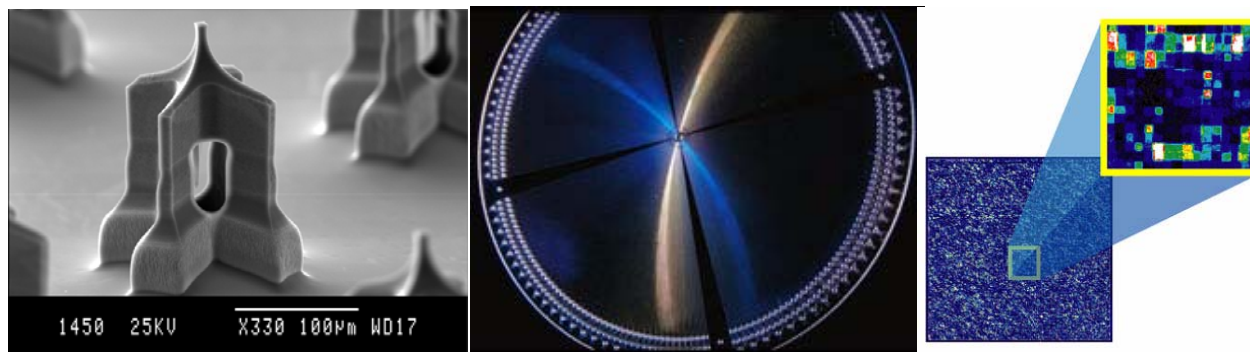


FIGURE 1. (a) BioMEMS microneedles for transdermal drug delivery [2], (b) Multiplex Lab-on-a-Chip DNA sequencer integrates 384 independently operatable capillary electrophoretic separation channels for high throughput analysis [3], (c) Gene expression microarray with > 1 million probes with zoom in of individual probe locations after hybridization reaction [4].

test as positive or negative.

BioChips use chemical reactions to identify unknown biomolecules. A library of probes are immobilized at specific locations on the chip surface to construct the BioChips. Fluorescently-labelled unknown bioagents react with these probes. The amount of fluorescence at each location is analyzed by image processing and reveals the relative quantitation of several thousand different nucleic acid transcripts. Bioinformatics is then used to extract genomic information.

For lithographically-fabricated probe arrays (e.g., Affymetrix), a couple of VLSI-like problems in probe placement and embedding [5][6] for DNA microarray BioChips design have been introduced. However, these problems are often dwarfed by the complexity of the more general bioinformatics problem of probe selection (deciding what probes should be placed on the chip) and image processing problem of the chip readout. An example of the importance of image processing can be garnered from Fig. 1(c), which shows a chip image after the unknown bioagents have reacted with the probe array. An alternative to lithographically constructing DNA probes directly on the chip involves synthesis of the DNA probes off-chip, and then inkjet based printing [7] or microfluidics-based probe transport to the desired location on the chip [8]. Microfluidics is also needed for BioChip fluid handling in high-throughput screening applications [9].

Due to the combination of the importance of microfluidics platforms and the similarity of microfluidic ICs to transistor ICs this report focuses solely on microfluidics-based Lab-on-a-Chips. LoC design issues range from synthesis and physical design to modeling and simulation-based verification. The next section in this report describes the technology and design issues for microfluidics-based chips. As with IC design, complexity is the primary motivating factor for CAD, as will be discussed in Section 2.2. Top-down design methodologies for handling this complexity are described in Section 3.1.

2.1. Lab-on-a-Chip

μ TAS and LoC are used interchangeably to describe chips that integrate common chemical or biological workbench processes, such as sample pretreatment and transportation, mixing, reaction, separation and detection [10]. The primary advantage of miniaturization advantages include the over 1000x reduction of sample consumption, leading to improved sensitivity and resolution of biomolecular detection. Two types of LoC platforms are being developed, categorized by their fluid transport mechanism.

2.1.1. Digital Microfluidics

Transport of immiscible gas bubbles, liquid droplets or particles (e.g., water droplets immersed in oil or air) have been explored as platforms for digital microfluidics. These platforms can be used as a discrete, randomly accessed, multi-analyte analyzer, in which reagents or samples, compartmentalized into droplets, can be processed in any desired order in parallel fashion. Of the wide variety of droplet/particle transport mechanisms (usually involving surface tension modulation [11]), the two that have found broad biofluidic applications include electrowetting on dielectric (EWOD) and dielectrophoresis (DEP).

In EWOD, the electrolyte droplet is placed on a hydrophobic surface. This surface is separated from the electrodes by a dielectric layer, as shown in Fig. 2(a) [12]. The droplet is moved by modulating the local surface tension around the droplet using an electric field. Droplets can be created from a reservoir (microfluidic digitization — Fig. 2(b)). Additional bio-processing functions such as droplet splitting (Fig. 2(b)), transportation (Fig. 2(c)), and merging (Fig. 2(d)) have been successfully demonstrated [13][14]. Recent applications include calorimetric enzyme-kinetic assays

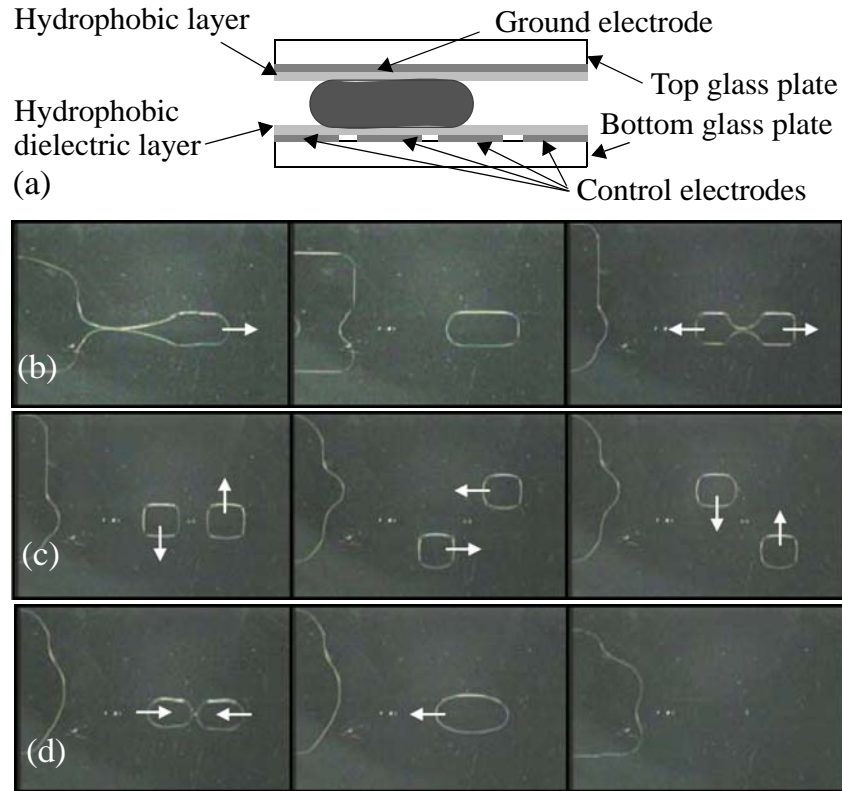


FIGURE 2. (a) Cross section of EWOD device [12]; (b) Droplet injection from reservoir (left) and split (right); (c) Independent and simultaneous droplet transport; and (d) two droplets are merged (left and center), and then merged with reservoir (right) [14].

[15] and on-chip sample processing for MALDI-Mass Spectroscopy [16].

DEP forces are induced on polarizable electrically neutral particles due to an induced dipole when subjected to a non-uniform electric field [17]. Such forces have been used widely to manipulate and separate cells [18][19].

2.1.2. Channel-based Microfluidics

Channel-based microfluidics [10] predates the earliest microarray or digital microfluidics concepts, and is still more widely used by researchers today because of its flexibility. For example, droplets have been formed using unstable two phase flows to enhance reaction. In terms of analogies to the IC world, digital microfluidics tends to use microarchitecture abstractions, while channel-based microfluidics tends to use analog circuit abstractions.

Channel-based LoC applications range from analysis of amino acids [20] and DNA [3] (exploiting biochemical separation), to chemical synthesis [21] (exploiting reaction). Chips integrating both reaction and separation have also been demonstrated [22].

Fluid flows in microchannels exploit two primary driving forces: pressure and electrokinetic (EK) forces. We focus on EK forces here because (1) they can be created by applying a voltage difference, like transistor-based ICs; (2) they can move biomolecules in a static bulk fluid, needed for molecular separation; and (3) they tend to be the predominant mechanism for complex LoC designs.

2.2. LoC Complexity

Multiplex complexity arises from arrayed integration of the same functional subsystem. If

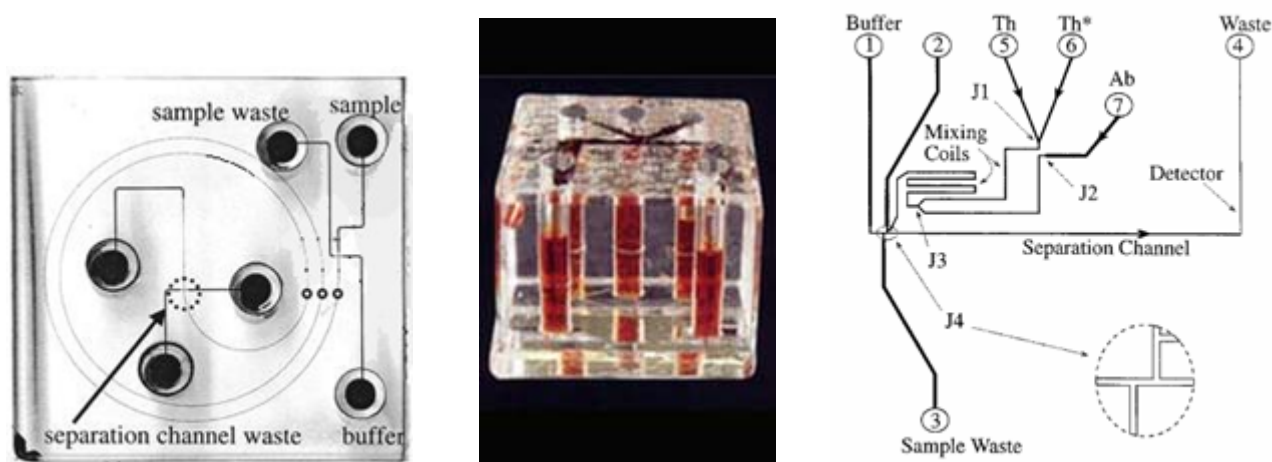


FIGURE 3. (a) Spiral Capillary Electrophoresis separation assay for amino acids [20]; (b) Microreactors for drug discovery [21]; (c) Multifunction immunoassay that integrates mixing, reaction, and separation [22].

the unit being arrayed is identical, e.g., the chip in Fig. 1(b), it is a homogeneous multiplex (analogous to RAM cells being arrayed into a memory IC). If the units have different sizes but the same function, then it is a heterogeneous multiplex. If it involves different functions, such as when mixing, reaction and separation are integrated as in the immunoassay in Fig. 3(c), then it is a multifunction system (analogous to custom ICs).

Computational fluid dynamics has been applied to channel-based [23][24] and to droplet-based [25] LoC simulation since the late-90's. This detailed simulation provides insight at the device level (e.g., for a single turn [26] — similar to device simulation in transistor-based ICs), but is impractical for system-level simulation. Typical system-level simulation runs can take days and require several GB of RAM. Additional challenges are enumerated in [27]. While fast solvers for gas flow have been developed [28], general purpose fast solvers are still not available.

3. Methods, Assumptions, Procedures

3.1. Top-Down Design Methodologies

To overcome these complexity limitations, researchers have been developing top-down approaches to handle complexity. Hierarchy is used to overcome the limitations of continuum simulations for channel-based LoCs. For example, the immunoassay in Fig. 3(c) can be partitioned into mixing, reaction, injection and separation subsystems. We call this the *functional decomposition* step (breaking the system into its functional elements similar to processor, memory and I/O in computer architecture). Each of these functional elements can then be hierarchically decomposed into straight channels, elbows, U-turns, as shown by the schematic in Fig. 4. This is the *geometrical decomposition* step [29].

The function-based categorization identifies the primary physical principles that need to be modeled to accurately and efficiently capture the element behavior. The geometry-based categorization allows use of parametric spatial integration to obtain geometrically parameterizable models, enabling a high degree of model reuse. This is in contrast to reduced-order models (ROMs) which can be obtained by applying nonlinear circuit model reduction techniques [30]. These ROMs have the potential to speed up simulation of a specific device, but need to be regenerated whenever the geometry changes, which is undesirable for iterative design.

When the fluidic network only involves electroosmosis, an analogy between bulk fluid and current flow can be used to obtain resistor-based circuit networks for fast and accurate simulation

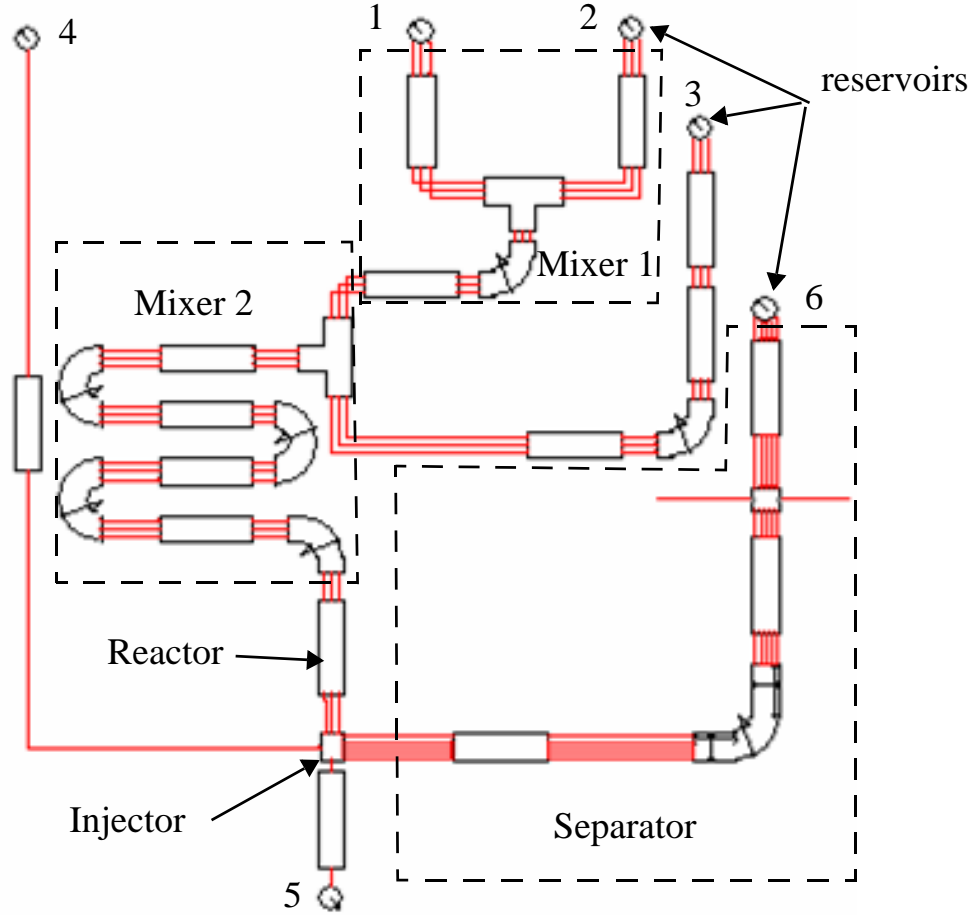


FIGURE 4. System-level Immunoassay schematic showing straight channels, U turns, elbow turns, and other elements used to integrate mixing, reaction, injection and separation functions on a single LoC [29].

[31]. This fluid transport model has been embellished with lumped parameter models for reaction, mixing, and separation for integrated LoC simulation [32].

A library for simulating both electrokinetic and pressure driven flows in microchannels as well as droplet transport is available commercially from Coventor as a fluidics library for their ARCHITECT simulator [33].

These first generation lumped approaches do not adequately capture dispersion sources in the lumped models. Separate continuum simulations are needed to determine the correction factors due to dispersion. When the primary dispersion mechanism is geometrical, this implies a numerical simulation run for every change in design geometry [34], making these approaches untenable for iterative geometrical design for LoC.

To overcome these limitations, an element library with nine mixing elements, ten separation elements, three reaction elements, and four injection elements has been developed at Carnegie Mel-

Ion University. This library has been used to create schematic networks that describe dozens of electrokinetically driven microfluidic mixers, separators, and integrated assays found in the literature [35].

3.2. Method of Moments

Two different methods were used to arrive at closed-form parameterized models for the physiochemical relationships that govern behavior in biofluidic microsystems. In essence these approaches aimed at replacing the partial differential governing equation. The first of these methods is the Method of Moments and is described here.

The species band concentration $c(y, z, t)$ within a separation channel is governed by the convection-diffusion equation [36]

$$\frac{\partial c}{\partial t} + u \frac{\partial c}{\partial z} = D \left(\frac{\partial^2 c}{\partial z^2} + \frac{\partial^2 c}{\partial y^2} \right) \quad (1)$$

where y and z are the transverse and axial positions respectively, t is the separation time from the channel entrance. The width of the species band can be characterized by variance, the square of the standard deviation of the average concentration profile c_m , which is defined as

$$\sigma^2 = \frac{\int_{-\infty}^{\infty} (z - \bar{z})^2 c_m \cdot dz}{\int_{-\infty}^{\infty} c_m \cdot dz} \quad (2)$$

where \bar{z} is the axial position of the species band's centroid in the channel.

Eq. (1) can be reformulated into a more tractable, reduced-dimension form in terms of spatial moments of the species concentration. Such moments are capable of describing the species band's main characteristics such as mass distribution, skew and variance without solving for detailed concentration distributions. We introduce a new coordinate frame, which moves at the species band's average velocity U , and normalize the equation to reduce all variables into dimensionless forms. Define a dimensionless axial coordinate ξ , transverse coordinate η and time τ by

$$\xi = (z - Ut)/w, \quad \eta = y/w, \quad \tau = Dt/w^2 \quad (3)$$

In terms of these dimensionless variables, Eq. (1) is rephrased to the following form in the concentration $c(\xi, \eta, \tau)$:

$$\frac{\partial c}{\partial \tau} = \frac{\partial^2 c}{\partial \xi^2} + \frac{\partial^2 c}{\partial \eta^2} - \text{Pe} \chi \frac{\partial c}{\partial \xi} \quad (4)$$

where $Pe = U_w/D$ is the Peclet number representing the ratio of convection and diffusive transport rates, and χ is the normalized species velocity relative to the average:

$$\chi(\eta) = (u - U)/U \quad (5)$$

We now recast Eq. (4) in terms of spatial moments of the species concentration. If the species band is entirely contained in the channel, Eq. (4) holds effectively over the axial domain $-\infty < \xi < \infty$ (the transverse domain is $0 < \eta < 1$), such that $c \rightarrow 0$ as $\xi \rightarrow \pm\infty$. Therefore, we can define spatial moments of the species concentration by

$$c_p(\eta, \tau) = \int_{-\infty}^{\infty} \xi^p c(\xi, \eta, \tau) d\xi, \quad m_p(\tau) = \int_0^1 c_p d\eta \quad (6)$$

Here, c_p is the p^{th} moment of the species concentration in the axial filament at η , and m_p is the p^{th} moment of the average concentration of the band, respectively. Note that as a consequence of the coordinate transformation Eq. (3), all moments are taken with respect to the moving frame (ξ, η) . For purposes of simulating analyte dispersion, it suffices to obtain the moments up to the second order. Specifically, c_0 provides the transverse distribution of the species mass in each axial filament within the channel and m_0 is the total species mass and can be chosen as $m_0=1$ without losing generality. Next, c_1 gives the axial locations of the centroid of the axial filaments in the species band, and hence indicates the skew of the band. Then, m_1 , the widthwise average of c_1 , is the axial location of the centroid of the entire species band in the frame (ξ, η) and always zero for this study [37]. Finally, m_2 can be used to determine the variance σ^2 of the species band by $\sigma^2 = w^2 (m_2/m_0 - m_1^2/m_0^2)$.

3.3. Reduced Order Modeling

The second approach to getting parameterized models is reduced order modeling. The tree in Fig. 5 distinguishes between numerical and analytical modeling methods. When problems cannot be solved analytically with closed-form solutions to the governing partial differential equations, additional approximations must be made. Numerical modeling makes these approximations in various ways. In full order methods, the original equations are discretized and solved in full form. This new system of equations can be very large and extremely computationally expensive to evaluate. Alternate numerical methods seek to reduce this computational complexity by removing nonessential degrees of freedom. These so called ‘reduced-order’ methods, such as state-space reductions,

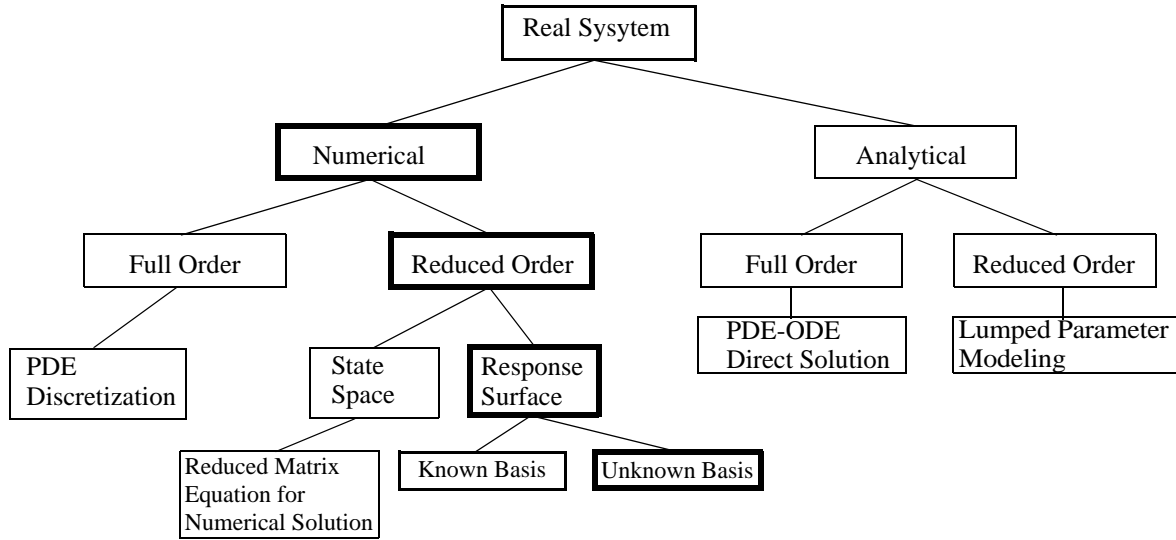


FIGURE 5. Related modeling methods.

reduce the scale of the discretization and eliminate unnecessary degrees of freedom in the discretization using basis reduction methods [38]. In other reduced order methods, if specific system responses are desired, such as the yield of a chemical reaction, a single function can be found through parameter estimation of a known functional form obtained from physical knowledge of the system [39]. Alternatively, if the system is a ‘black-box’, the basis of its functional representation is unknown due to a lack of physical knowledge. In these cases, non-linear regression techniques, such as the neural network can be used where the choice of functional form is not necessary [40][41][42]. The focus of the modeling in this work is to create response surface models targeted at design synthesis. The developed models are valid only within the specified design space, which may or may not be rectangular.

3.3.1. Neural Network Reduced Order Model

The creation of a model using numerical simulations involves the definition of a mapping between the model inputs and parameterized outputs. The numerical simulation, which approximates the physics of the real model, is used to sample and define this mapping, followed by a multi-dimensional regression to create a closed-form model. This model approximates the device's actual performance mapping function (the real model). A neural network is a robust regression tool for identifying this mapping because it is efficient with sparsely sampled data, can identify mappings without any prior information about the desired form of the model or physics, and results in a closed-form function parameterized by the weights inherent to the connections between the nodes

of the NN. In this work all calculations are performed in MATLAB [43].

When the nodes of a NN are arranged in layers, with connections only between adjacent layers, the topology is referred to as feed-forward. An example of this topology is shown in Fig. 6. The NN operates by passing the data at its input nodes through a series of weights to the nodes in the layers above. Each node in the layer takes the weighted sum of the output of the nodes below as the argument of some arbitrary bound non-linear, or unbound linear transfer function. The use of non-linear transfer functions in all nodes between the input and output layers and linear transfer functions at the output nodes gives the NN the useful property of being a universal function approximator when at least two of these hidden layers are present, and sometimes only one layer when the approximated function is bound [44]. This means given enough nodes in the hidden layers, the NN will be able to approximate a function of any degree of non-linearity, with a finite number of discontinuities, whether bound or otherwise, to an arbitrary degree of accuracy [45]. So with a certain degree of confidence, it can be assumed that the NN regressor will provide a robust method for mapping the performance function for a very large class of device models, given sufficient sampling data to represent the details of the underlying structure of the mapping.

To uniquely define the NN model, the number of layers, nodes per layer, node transfer function, and interconnecting weights must be specified. In general, the number of layers and node transfer functions are chosen to achieve the universal approximator criterion, namely two hidden layers with bound non-linear transfer functions and linear transfer functions on the output nodes. In

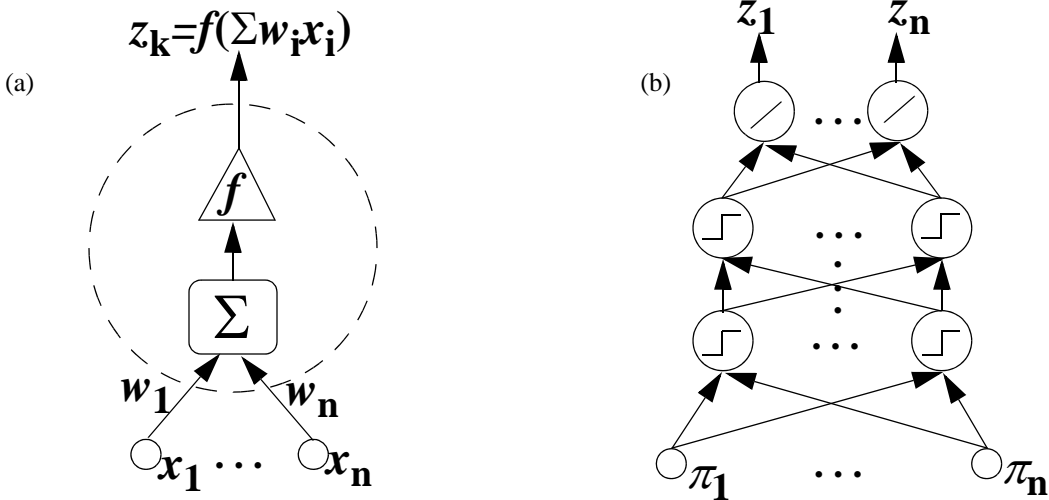


FIGURE 6. (a) The activation, z_k , of a node is a function of the weighted inputs of the previous layer's activations. (b) General feedforward neural network topology utilizing multiple hidden layers with an unspecified number of nodes with non-linear activation functions, and a single output node using a linear activation function.

some of the models, namely for the injectors, only a single hidden layer of nodes is required since the targeted performance functions are almost strictly monotonic and bound. The weights connecting the nodes are chosen so as to minimize an error criterion, such as the mean squared error of the NN model to the sampled mapping data. This process is called training the neural network. To choose the number of nodes in the hidden layers, the number of nodes is varied while the NN trained. The number of nodes which minimizes the degree of underfitting and overfitting is chosen for the final model.

Underfitting and overfitting manifests itself through the bias and variance of the network as a function of the complexity of the network (more nodes makes a more complicated network). The mean squared error decomposition for a particular data point, x , for noiseless source data leads to the following metric for the networks estimate:

$$MSE(\hat{f}(x)) = E[(\hat{f}(x) - E[\hat{f}(x)])^2] + (E[\hat{f}(x)] - f(x))^2 \quad (7)$$

where $\hat{f}(x)$ is the network's estimate for the mapping at point x , $f(x)$ is the actual target data at that point, and $E[...]$ represents the expectation for an ensemble of training data sets. The first summand is a measure of the variance of the estimate at that point, and the second summand is a squared measure of the bias, or difference between the expected estimate and the actual data at that point, for sets of training data [46][47].

Networks of lower complexity (fewer nodes) will tend to have a high bias and a low variance, whereas networks with high complexity (more nodes) will tend to have low bias and high variance. To find the optimal complexity for a model, the number of nodes in the hidden layer is chosen so as to find the best trade-off between bias and variance. Two types of error measurements are conducted during the construction of a NN: 1) Training error measures the error of the network to the original data set (the training data set). Since the training process is an optimization problem that constrains the network to have a low bias on this data set, this is not a good indicator of the overall generalized error of the network (for example, it doesn't indicate the bias or variability of the network to data it has not seen in the training data set). 2) To measure the variance of the network, a different test data set must be used, consisting of points not included in the training data set, which provides a measure of the variability at points where the network was not constrained during its learning process. This is a more generalized error as indicated by Eq. (7). A network with a high bias or variance will produce a large generalization error, and this is the error that must be mini-

mized to choose the optimal complexity of a network. Efficient methods for measuring this generalization error on test data sets is outline in the next section.

3.3.2. Model Selection

A technique referred to as k-fold cross validation (KFCV) is a practical method for measuring the generalized error in Eq. (7). This technique is efficient since it does not require a separate validation set. All data is used in the training of the neural network. As shown in Fig. 7, this method randomly sorts the training data set into k subsets (folds). The network is trained on a set consisting of $k-1$ subsets and trained on the one remaining subset. In order to avoid biasing, each subset plays the role of the validation set exactly once. The errors measured for each validation subset iteration are then averaged together to provide an unbiased measure of the generalized error. It has been determined that k of 5 or 10 provides the best estimate of the generalization error [38][39].

Other types of validation include, the simple test set validation. In this method, a fixed number of data points are removed from the acquired training data and are used after training to measure the ability of the model to generalize to unknown data. This method is undesirable for several reasons. First, it reduces the amount of data available for training, which is important when the data set is not large to start. Second, it has a large bias error since the removed data is a fixed set that clearly only significantly affects the accuracy of the model in the region of missing data. It is also unclear how to decide which data points should be removed from the original set.

Leave-one-out cross validation improves on the test set validation method by training a

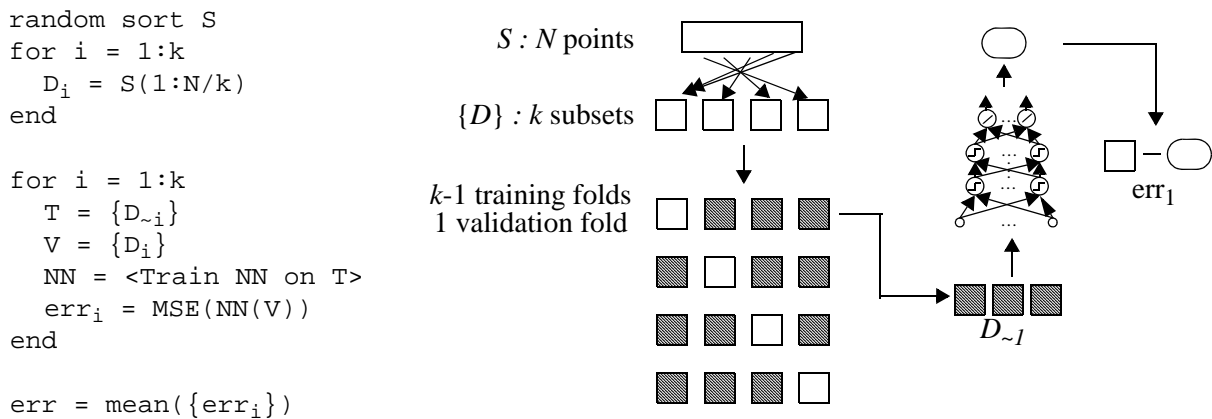


FIGURE 7. k-fold cross validation algorithm. First divide the N points in the dataset into k subsets (3 to 5 subsets is typical). Then, identify one subset (i) as the validation set, used to compare with the trained neural network. The other $k-1$ subsets ($\sim i$), are used to train the neural network. Each subset is the validation set exactly once. The error for each iteration is averaged at the end to provide an estimate of the generalization error for the network. This is an efficient use of data, since all data is used for training *and* validation. There is no exclusive validation set.

model N times for a data set with N points, leaving one point out of the training data set each time, and measuring the error of the model on that point afterwards, [41]. This removes the bias introduced from using a fixed test set, but since each of the N training data sets are very similar since only one point is missing, the variance of the resulting model can be very high.

3.3.3. Design of Experiments

The neural network can only be as accurate as the data used for training. There are several factors to consider when choosing data to train the network, these and related issues are generally described as *design of experiments* or *data mining* [46]. First is the number of total data points to provide. There is no fixed number of samples that can be used in general, since it depends on the number of dimensions and the complexity of the function in the space. However, rapidly changing functions in high dimensions will require many more samples than slowly changing functions in low dimensions. As an example, for the injector models in the following section with four dimensions, on the order of 1000 points are used. Second is where to sample in the space spanning the dimensions of the function to be learned. If knowledge of the underlying function shape is used, samples can be increased near points of rapid change. Otherwise, uniform sampling throughout the space is used. In order to maximize the number of samples per dimension, it is best to sample the data not on a uniform grid. Fig. 8 shows an example in two dimensions where the number of samples, n_s , is taken on- and off- grid. When taken on-grid there are $\sqrt[k]{n_s}$ unique points per dimension (in an k -dimensional space), where off-grid results in n_s unique points per dimension.

The off-grid results can be generated using pseudo-random sequence generators, which asymptotically have a lower discrepancy (defined in a moment) than purely random sequences. The benefit of using a grid to sample is that the points are equidistantly separated eliminating all bias in

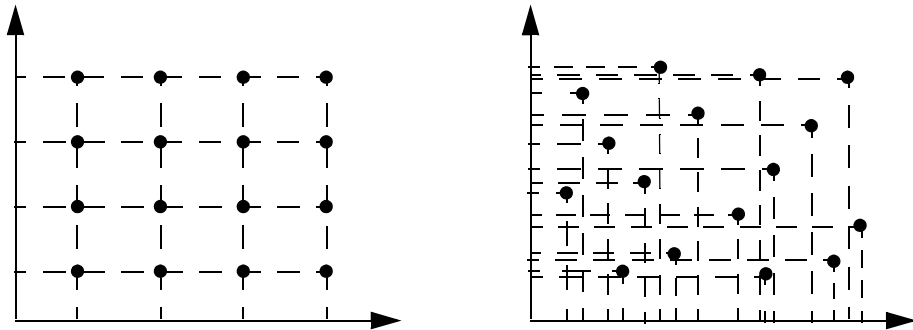


FIGURE 8. (a) Uniform sampling on a grid in two-dimensions with a total of 16 data points resulting in effectively 4 data points per dimension. (b) Uniform sampling off-grid in two dimensions, again with 16 data points, resulting in effectively 16 points per dimension, a better more denser sampling per dimension.

the sampling sequence -- one area isn't over-sampled while another area with important structure might be undersampled. Unfortunately, as mentioned before, the use of a grid reduces the number of unique samples per dimension. This trade-off can be balanced by using a low-discrepancy sequence that maximizes the sampling per dimension, while minimizing the clustering of high-discrepancy sequences. These low-discrepancy sequences can be achieved asymptotically using pseudo-random sequences, such as the common Sobol [48][49], Faurer [50], Halton [51], Hammersley [52], or Niederreiter [53] generators. The quasi-random sequences were originally created to make more rapidly converging high-dimensional integrals using Monte-Carlo evaluation, but the results apply to neural networks, as well.

Discrepancy can be defined on the unit hypercube as the maximum difference between the volume (for any volume defined within the hypercube -- with one vertex at the origin for the star-discrepancy) and the average number of points included within the volume,

$$D^* = \sup_{\{x\} \in [0,1)^d} \left| \frac{1}{N} \sum_{i=1}^N \prod_{j=1}^d (1 - \Theta(a_{ji} - x_j)) - \prod_{j=1}^d x_j \right| \quad (8)$$

where Θ is the unit step function, a_{ji} is the i^{th} sampled point out of N in the j^{th} dimension out of d , and x_j represents the edge in the j^{th} dimension of the variable cube measuring enclosed points within the unit hypercube. A sequence $\{\bar{a}_j\}$ is considered uniformly distributed if the average number of points within a volume equals the size of the volume as the sequence becomes infinite (i.e. uniform distributions have all possible volumes “completely” full with points, whereas non-uniform distributions will leave some volumes more, or less, full).

For this work, the non-asymptotic nature of the datasets, with on the order of 1000 points, means the discrepancy of pseudo-random sequences are not significantly better than a randomly perturbed deterministic uniform sequence. Further, the randomly perturbed deterministic sequence has several additional post-processing benefits not possible with quasi-random sequence. For example, if certain regions of the problem domain are marked as infeasible, the perturbed sequence can be generated so as to “push” points out of the infeasible region. This prevents running simulations on points that will eventually be excluded from the model. Such perturbations can be applied to a quasi-random sequence, but only after the sequence has been generated as a whole, thus such perturbations negate the low-discrepancy benefits of using such sequences in the first place.

The improvement in training accuracy for scattered data points rather than uniform gridded

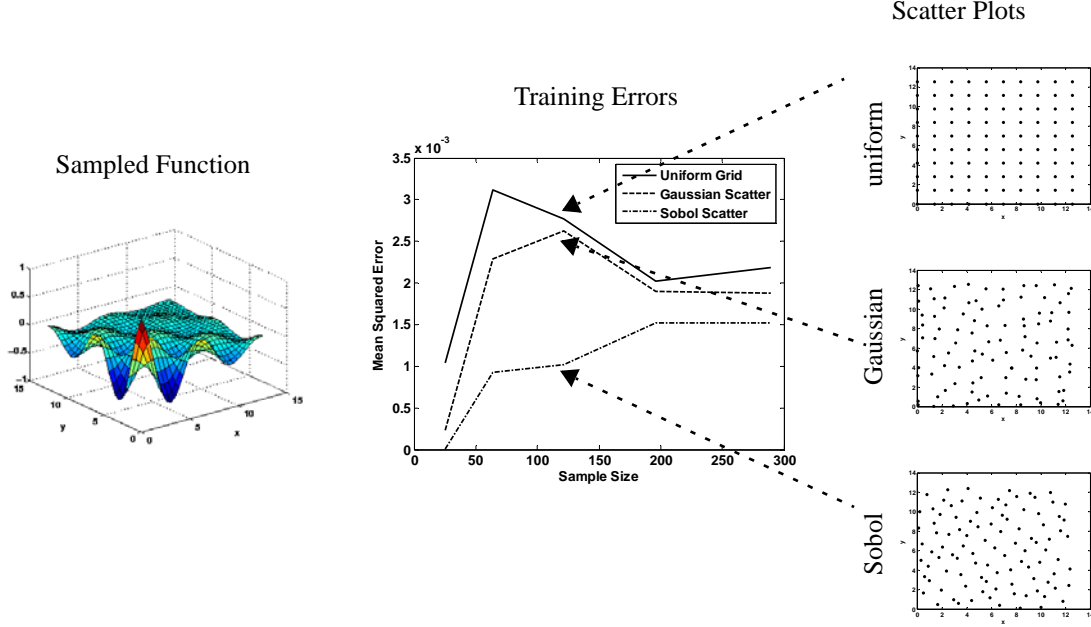


FIGURE 9. A neural network is trained on the function shown on the left. The network has a single hidden layer of tansig neurons and one linear output layer. It is trained for 450 max iterations, and the sample size is varied. The error measured is the mean squared error of the network to the training data (not a measure of the generalization error). The Sobol quasirandom sequence is the best performing, followed by the Gaussian sequence, and finally the uniform grid. The Gaussian sequence is a simpler algorithm and is compatible with domain shaping methods.

points, can be shown with an example. In this example, three methods are compared for training a NN on a two-dimensional function:

$$f(x, y) = \cos(x) \sin(y) \exp\left(\frac{-x-y}{2\pi}\right) \quad (9)$$

First, x and y are sampled from 0 to 2π on a uniform grid, using the `meshgrid()` command in MATLAB. Then x and y are sampled from 0 to 2π with each point perturbed off the uniform grid by an amount determined by a gaussian with a standard deviation equal to 3% of the domain size (2π). In this case, if any points go outside the defined domain, they are resampled until they fall within 0 to 2π . Finally, x and y are sampled from 0 to 2π using the Sobol pseudo random sequence generator [48][49][54], obtained from [55]. Using each of these sampling methods, a NN is trained and the error to the training data is measured and plotted in Fig. 9. The neural network has the topology of Fig. 6b with a single hidden layer of 10 neurons with a tansig transfer function,

$$f(x) = \frac{2}{1 + \exp(-2x)} - 1 \quad (10)$$

and a single output layer with a linear transfer function,

$$f(x) = x \quad (11)$$

The network is trained for a maximum of 450 iterations using the Levenberg-Marquardt algorithm [56]. Since the initialization of the algorithm is random, each point in the graph of Fig. 9 represents the average of 40 repeated training experiments. The initial slope in the individual traces is most likely due to the oversimplified representation of Eq. (9) from undersampling with the smaller datasets. As the dataset increases, the additional structure of Eq. (9) makes the training more difficult leading to the initial rise in error as it progresses to some other asymptotic value that depends on the complexity of the network. It can be concluded from Fig. 9 that the Sobol pseudo random sequence provides the best results, followed by the Gaussian sequence, and finally the uniform sequence. Due to the more complex algorithm and loss of the low-discrepancy qualities when the points are moved to avoid infeasible areas of the parameter space, the Gaussian algorithm will be the default algorithm used, unless otherwise noted.

4. Results, Discussion

4.1. Component Models Development and Verification

The goal of each behavioral model is to capture the input-output signal flow relationship of the pin values that define biofluidic state at the inlet and outlet of each element [57]. This captures the physical phenomena being modeled in that element. In addition, an electrical resistance is associated with each element to relate the EK current flow through the element to the inlet and outlet voltages. In contrast to the bottom-up reduced-order model approaches, our behavioral models possess several important attributes to enable accurate and efficient system-level simulations of complex LoCs. Our analytical models effectively account for the same multi-physics (e.g., electrostatics, fluidics and mass transfer) as numerical simulation tools. They do not require any parameters from user-conducted experiments or numerical simulations to capture interactions between the elements, and hence provide seamless model interconnectivity. Most importantly, they are in closed-form and are all parameterized by element dimensions and material properties; therefore they are reusable, fast to evaluate, and well suited for an iterative simulation-based design methodology.

As discussed above, depending on the physical phenomena of individual devices, contents of the behavioral model libraries will be different. Hence, models for the micromixers and electro-

phoretic separation systems will be developed separately, and are available in separate model libraries for schematic-based simulations.

4.1.1. Electrokinetic Passive Micromixers

The EK passive micromixer library consists of models for nine elements, which includes reservoirs (sample and waste), slightly tapered straight mixing channel, turns (90° or 180° , clockwise or counter-clockwise), as well as converging and diverging intersections. In this section, we will present behavioral models for basic elements such as the slightly tapered mixing channel, converging and diverging intersections. Other elements can be modeled in a similar fashion.

Slightly Tapered Straight Mixing Channel. The tapered straight mixing channel, in which different samples a and b mix with each other, has one inlet and one outlet, with different cross-sectional area. It is critical in designing a geometrical focusing micromixer [58]. Electrically, it is modeled as a resistor and the resistance is given by

$$R = \int_0^L \frac{dz}{w(z)h(z)C_e} \quad (12)$$

where w and h are the channel width and depth (both are functions of the axial coordinate z), C_e is the electric conductivity of the buffer solution in the channel. As a special case, within a straight channel with the uniform cross-section, Eq. (12) can be reduced to

$$R = \frac{L}{hwC_e} \quad (13)$$

To obtain the sample concentration profile at the outlet, we partition the slightly tapered straight channel into a series of segments (segment number tends to infinity), each with uniform cross section. In each segment, the convection-diffusion equation is solved to establish the input-output relationship of concentration coefficients between the inlet and outlet of the segment. Then all the segmental solutions are multiplied and the concentration coefficients $d_n^{(out)}$ ($n = 0, 1, 2, \dots$) at the channel outlet are attained as [59]

$$d_n^{(out)} = d_n^{(in)} e^{-\gamma n^2 \pi^2 \frac{LD}{E_{in} \mu w_{in}^2}} \quad (14)$$

where $d_n^{(in)}$, w_{in} and E_{in} are the concentration coefficients, channel width and electric field at the inlet, γ is a factor capturing the effect of the cross-sectional shape on mixing [59], D and μ are the diffusivity and electrokinetic (including both electroosmotic and electrophoretic) mobility of the sample

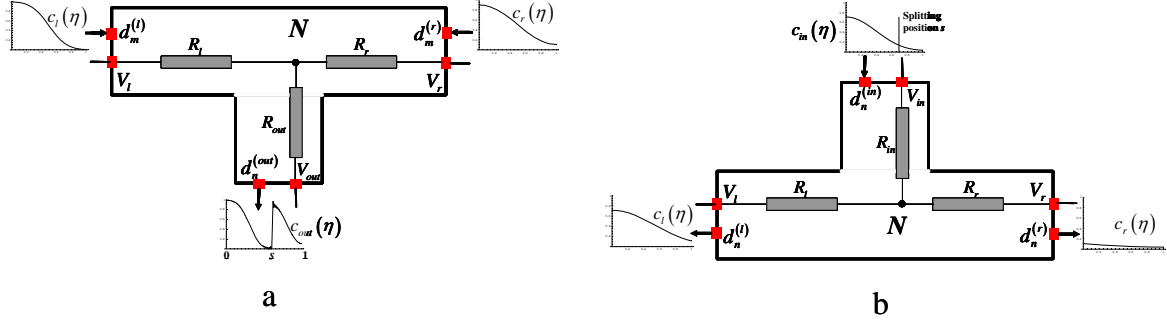


FIGURE 10. (a) Behavioral model structure for the converging intersection in the micromixer. Index l, r and out represent the quantities at the left inlet, right inlet and outlet. Both electric (V) and biofluidic (d_m) pins are defined at the terminals of the model. Electrically, it is modeled as a combination of three resistors (R_l, R_r and R_{out}) with zero resistance. Different sample concentration profiles, $c_l(\eta)$ and $c_r(\eta)$, at inlets are merged and compressed at the outlet $c_{out}(\eta)$. (b) Behavioral model structure for the diverging intersection in the micromixer. Similarly, index l, r and in represent the quantities at the left outlet, right outlet and inlet. Sample concentration profile at inlet, $c_{in}(\eta)$ is split and stretched out into two parts, $c_l(\eta)$ and $c_r(\eta)$, flowing out of the outlets.

and L is the channel length. The special case of a straight channel with the uniform cross-section yields $\gamma = 1$.

Converging Intersections. Fig. 10 shows the behavioral model structure of converging and diverging intersections used in micromixers [60]. Arrows at pins indicate the direction of signal flow for computing biofluidic pin values and state. The converging intersection has two inlets and one outlet, and acts as a combiner to align and compress upstream sample streams of an arbitrary flow ratio s (defined below) and concentration profiles side-by-side at its outlet (Fig. 10a). As its flow path lengths are negligibly small compared with those of mixing channels, such an element can be assumed to have zero physical size, and electrically represented as three resistors with zero resistances between each terminal and the internal node N ,

$$R_l = R_r = R_{out} = 0 \quad (15)$$

Here, N corresponds to the intersection of flow paths and the subscripts l, r and out represent the left and right inlets, and the outlet, respectively. Defining $d_m^{(l)}$ and $d_m^{(r)}$ ($m=0,1,2,\dots$) as the Fourier coefficients of the sample concentration profiles at the left and right inlets respectively, then the coefficients $d_n^{(out)}$ ($n=0,1,2,\dots$) of the profile at the outlet ($c_{out}(\eta)$) are related to $d_m^{(l)}$ and $d_m^{(r)}$ by,

$$c_{out}(\eta) = \sum_{n=0}^{\infty} d_n^{(out)} \cos(n\pi\eta) = \begin{cases} \sum_{m=0}^{\infty} d_m^{(l)} \cos\left(\frac{m\pi\eta}{s}\right) & , \quad 0 \leq \eta < s \\ \sum_{m=0}^{\infty} d_m^{(r)} \cos\left(\frac{m\pi(\eta-s)}{1-s}\right) & , \quad s \leq \eta < 1 \end{cases} \quad (16)$$

Eq. (16) shows that the concentration profile at the outlet can be treated as a superposition of the scaled-down profiles from both inlets, where $s = q_l/(q_l + q_r) = I_l/(I_l + I_r)$ denotes the interface position (or flow ratio, the ratio of the left flow rate q_l to the total flow rate $(q_l + q_r)$) between incoming streams in the normalized coordinate at the outlet (note that flow rates q_l and q_r are respectively linear with the electric currents I_l and I_r).

Solving Eq. (16) yields $d_n^{(out)}$ as,

$$\begin{cases} d_0^{(out)} = d_0^{(l)} s + d_0^{(r)} (1-s) \\ d_{n>0}^{(out)} = s \sum_{m=0}^{m \neq ns} d_m^{(l)} \frac{f_1 \sin(f_2) + f_2 \sin(f_1)}{f_1 f_2} + s \sum_{m=0}^{m=ns} d_m^{(l)} + (1-s) \sum_{m=0}^{m=n(1-s)} (-1)^{n-m} d_m^{(r)} \\ \quad + 2(-1)^n (1-s) \sum_{m=0}^{m \neq n(1-s)} d_m^{(r)} \left(\frac{\cos(F_2/2) \sin(F_1/2)}{F_1} + \frac{\cos(F_1/2) \sin(F_2/2)}{F_2} \right) \end{cases} \quad (17)$$

where $f_1 = (m - ns)\pi$, $f_2 = (m + ns)\pi$, $F_1 = (m + n - ns)\pi$ and $F_2 = (m - n + ns)\pi$. Since the sample concentration profiles at the inlets are scaled down, the Fourier series mode at the inlets is not orthogonal to that at the outlet. Therefore, the calculation for the coefficient of a certain Fourier mode at the outlet also depends on the other modes at the inlets.

Diverging Intersections. The diverging intersection has one inlet and two outlets and is the dual of the converging intersection. It splits the incoming flow and electric current into two streams that exit from the outlets. It can also be represented by three zero-resistance resistors,

$$R_m = R_l = R_r = 0 \quad (18)$$

where subscripts in , l and r represent quantities at the inlet, the left and right outlets.

Defining $d_m^{(in)}$ ($m=0,1,2,\dots$) as the Fourier coefficients of the sample concentration profile at the inlet. Then the coefficients at the left and right outlets $d_n^{(l)}$ and $d_n^{(r)}$ are respectively given by

$$\begin{cases} d_0^{(l)} = d_0^{(in)} + \sum_{m=1}^{\infty} d_m^{(in)} \sin(\phi_1) / \phi_1 \\ d_{n>0}^{(l)} = 2 \sum_{m=0}^{m \neq n/s} d_m^{(in)} (-1)^{n+1} \phi_1 \sin(\phi_1) / f_1 f_2 + \sum_{m=0}^{m=n/s} d_m^{(in)} \end{cases} \quad (19)$$

and

$$\begin{cases} d_0^{(r)} = d_0^{(in)} - \sum_{m=1}^{\infty} d_m^{(in)} \sin(\phi_1) / \phi_2 \\ d_{n>0}^{(r)} = 2 \sum_{m=0}^{m \neq n/(1-s)} d_m^{(in)} \phi_2 \sin(\phi_1) / F_1 F_2 + \sum_{m=0}^{m=n/(1-s)} (-1)^{m-n} d_m^{(in)} \end{cases} \quad (20)$$

where $f_1 = (n - ms)\pi$, $f_2 = (n + ms)\pi$, $F_1 = (n + m - ms)\pi$, $F_2 = (n - m + ms)\pi$, $\phi_1 = ms\pi$ and $\phi_2 = m(1 - s)\pi$. Similar to the converging intersection, s is the normalized splitting position (or ratio).

It should be pointed out that in contrast to the resistor-based mixing models [32][60] that take advantage of the analogy between fluidic and sample transports and only convey the average concentration values through the entire network, our models (Eqs. (14), (17), (19) and (20)) propagate sample concentration profiles characterized by the Fourier series coefficients. This removes the requirement of complete mixing (along channel width) at the end of each channel in the network imposed by the resistor-based model and allows for optimal design of both effective and efficient micromixers.

4.1.2. Electrophoretic Separation Chips

The electrophoretic separation library includes models for ten basic elements: turns (90° or 180°, clockwise or counter-clockwise), straight channel, detector, injector, injection channel and reservoirs (sample and waste). In this section, behavioral models for basic elements such as separation channels (straight and turn) will be developed to analyze the band-spreading effect caused by molecular diffusion and turn dispersion. Additionally, a detector model applicable for both DC and transient analysis will be presented. Models of the other elements can be derived using the same principles.

Fig. 11 shows the behavioral model structure of electrophoretic separation channels (straight or turn). Arrows indicate the direction of signal flow for calculating biofluidic pin values and state. Electrically, separation channels are modeled as resistors in the same way as the uniform straight mixing channels (for a constant-radius turn, L in Eq. (13) is replaced by $L = r_c \theta$, where r_c and θ are the mean radius and angle included by the turn, see [37][59] for detailed geometrical interpretation).

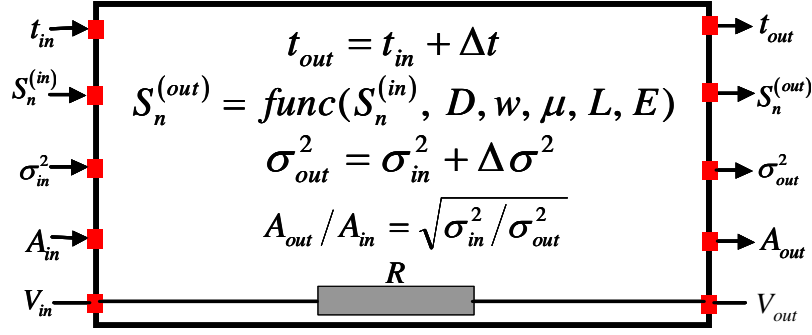


FIGURE 11. Behavioral model structure for separation channels in electrophoretic separation microchips. Index *in* and *out* represents the quantities at the inlet and outlet. Both electric (*V*) and biofluidic (*t*, *S_n*, *σ²* and *A*) pins are defined at the terminals of the model. Electrically, the channel is modeled as a resistor (*R*). The variations of biofluidic pin values due to dispersion effects are captured by the model.

Additionally, symbols and characters used in this section are defined the same as those for the mixer, unless otherwise noted. The residence time Δt of a species band within a separation channel (the time for the band to move from the channel inlet to outlet) is given by

$$\Delta t = \frac{L}{\mu E} \quad (21)$$

The calculation of changes in the skew coefficients and variance depends on the specific element [37] and the inherent variable is the residence time Δt obtained by Eq.(21). For a straight separation channel,

$$S_n^{(out)} = S_n^{(in)} \cdot e^{-(n\pi)^2 \Delta t D / w^2} \quad (22)$$

$$\sigma_{out}^2 - \sigma_{in}^2 = \Delta \sigma^2 = 2D \cdot \Delta t \quad (23)$$

For a separation turn,

$$S_n^{(out)} = \begin{cases} \pm \frac{8\theta w^2 \left(1 - e^{-(n\pi)^2 \Delta t D / w^2}\right)}{(n\pi)^4 \Delta t D} + S_n^{(in)} e^{-(n\pi)^2 \Delta t D / w^2}, & n = 1, 3, 5 \dots \\ S_0^{(in)}, & n = 0 \end{cases} \quad (24)$$

$$\begin{aligned} \Delta \sigma^2 = 2D\Delta t \pm \frac{8w^4\theta}{D\Delta t} \sum_{n=1,3,5\dots}^{\infty} \left(\frac{S_n^{(in)} \left(1 - e^{-(n\pi)^2 D\Delta t / w^2}\right)}{(n\pi)^4} \right) \\ + \frac{64w^6\theta^2}{(D\Delta t)^2} \sum_{n=1,3,5\dots}^{\infty} \frac{\left(-1 + e^{-(n\pi)^2 D\Delta t / w^2} + (n\pi)^2 D\Delta t / w^2\right)}{(n\pi)^8} \end{aligned} \quad (25)$$

where subscripts/superscripts *in* and *out* represent quantities at the inlet and outlet of the channel. In Eqs. (24) and (25), the “+” sign is assigned to the first turn and any turn strengthening the skew caused by the first; the “-” sign is assigned to any turn undoing the skew from the first.

Joule heating induced dispersion models were also developed in this project [61], however, are not presented for the sake of conciseness. They are more often than not used to verify that the electric fields are not too high to cause additional band broadening.

Assuming a Gaussian distribution of the average concentration c_m of the species band at element terminals always, we can obtain the amplitude of the species band by

$$A_{out}/A_{in} = \sqrt{\sigma_{in}^2/\sigma_{out}^2} \quad (26)$$

For the detector model, the variance change associated with the detector path length L_{det} is given by [62]

$$\Delta\sigma^2 = L_{det}^2/12 \quad (27)$$

4.1.3. Model Implementation

To demonstrate use of the above parameterized models for top-down designs, we have implemented the models in the Verilog-A analog hardware description language, and in Matlab. In the Cadence implementation, symbol views for each of the elements are used to compose a schematic within Cadence’s [63] integrated circuit design framework. The Cadence design framework is used to automatically netlist the complex topologies in the biofluidic LoC schematics, and Spectre is used as the simulator. Similar tools from other vendors, or custom schematic entry tools and solvers that can handle both signal flow and Kirchhoffian networks could have been also used.

An important issue of implementing separation channel models of turn geometry (Eqs. (24) and (25)) is the real-time determination of the turn “sign”. Providing this flexibility allows a single turn symbol to be used for constructing arbitrary topologies such as a serpentine, spiral or their combination thereof as will be shown later. To address this, two sets of flags are used in the models. One is the system flag F_s , stored as the zero-*th* component of the skew coefficients ($S[0]$ in Table 1) to record the direction of the skew caused by the first turn or elbow. The other is the intrinsic flag F_i of individual elements. For example, $F_i = 1$ is for turns or elbows involving clockwise flow of species bands; $F_i = 2$ is for counter-clockwise turns or elbows. Since straight channels do not incur

Table 1. Definition of Biofluidic Pins

| Micromixing | | |
|----------------------------|----------------------------|---|
| Bus | Pins connected | Description |
| $d [0:29]$ | Concentration coefficients | $d [0:9]$: the 1 st sample, $d [10:19]$: the 2 nd , $d [20:29]$: the 3 rd |
| Electrophoretic Separation | | |
| Bus | Pins connected | Description |
| $t [0:2]$ | Separation time | $t[0]$ for the 1 st species, $t[1]$ the 2 nd , $t[2]$ the 3 rd |
| $\sigma^2 [0:2]$ | Variance | $\sigma^2 [0]$ for the 1 st species, $\sigma^2 [1]$ the 2 nd , $\sigma^2 [2]$ the 3 rd |
| $A [0:2]$ | Amplitude | $A [0]$ for the 1 st species, $A [1]$ the 2 nd , $A [2]$ the 3 rd |
| $S [0: 30]$ | Skew coefficients | $S [0]$: the direction of the skew caused by the 1 st turn $S [1:10]$: the 1 st species $S [11:20]$: the 2 nd , $S [21:30]$: the 3 rd |

any skew, no flag is needed. During simulations, $F_s = 0$ (i.e., $S[0]=0$) is first generated by the injector, which is the most upstream element of a separation channel and hence initiates the computation of the separation state. Then as the species band migrates to the first turn or elbow, F_s is irreversibly set to the intrinsic flag F_i of them. Afterwards, the written F_s is compared with F_i of each downstream element as the band moves on. If they are identical, a “+” sign is used for the element, otherwise a “-” sign. Fig. 12 shows the codes for a turn involving clockwise flow of species bands to implement this logic and determine the sign.

4.1.4. Schematic-Based Simulation

In this section, we will first describe the simulation procedure, in which the Kirchhoff’s resistor network to predict electric current and field as well as the signal flow network to evaluate biofluidic state values (e.g., steady-state mixing concentrations and transient electrophoretic species band shapes) are solved sequentially. Then, the results of schematic simulations exploring various micromixers and separation microchips will be discussed and validated with numerical and experimental data.

Simulation Description. Schematic simulations for mixers and separation chips involve both electric and biofluidic calculations. For DC analysis, given the applied potential at reservoirs, system topology and element dimensions, nodal voltages at element terminals within the entire system are first computed by Ohm’s and Kirchhoff’s laws using the resistor models presented in the last section.

```

`define NUM 10                                // The number of terms in the Fourier series
...

module Uturn1(...);                          // Module declaration
parameter real sp_D_0 = 500;                 // Diffusivity of the zero-th species; unit:  $\mu\text{m}^2/\text{s}$ 
parameter real R2 = 110 ;                    // Outer radius of the U shape turn; unit:  $\mu\text{m}$ 
parameter real R1 = 100 ;                    // Inner radius of the U shape turn; unit:  $\mu\text{m}$ 
...

integer Fs;                                  // System flag
integer Fi;                                  // Intrinsic flag
real dummy1;                                 // Intermediate variable
real W;                                       // Channel width
real delta_t_0;                              // The residence time of the zero-th species within the turn; unit: s
real pi;                                     //  $\pi$ 
...

analog begin                                // Behavioral description begins

    Fi = 1;                                  // The present turn is clockwise
    W = abs (R2-R1);
    Fs = SKA(sk_in[0]);                      // sk_in is the skew coefficient; sk_in[0] stores the system flag
    pi = 3.1415926;
    ...

    if (Fs == 0) begin                       // system flag is not set yet; no turns or elbows upstream
        SKA(sk_out[0])<+(Fi);               // set the system flag to the intrinsic flag
        generate i (1, `NUM, 1) begin       // this loop is to calculate skew coefficient
            dummy1 = 1- exp(-(2*i-1)*(2*i-1)*pi*delta_t_0*sp_D_0/(W*W));
            Part1_Sk = 8*pi*W*W*dummy1/(pow((2*i-1), 4)*pow(pi, 4)*delta_t_0*sp_D_0); // the 1st part in Eq. (13), use "+"
            ...
        end
        ...
    end

    end else if (Fs == 1) begin              // Species flow in the first turn is clockwise
        SKA(sk_out[0])<+SKA(sk_in[0]);       // Convey the system flag
        generate i (1, `NUM, 1) begin
            dummy1 = 1- exp(-(2*i-1)*(2*i-1)*pi*delta_t_0*sp_D_0/(W*W));
            Part1_Sk = 8*pi*W*W*dummy1/(pow((2*i-1), 4)*pow(pi, 4)*delta_t_0*sp_D_0); // use "+" sign
            ...
        end
        ...
    end

    end else begin                          // Species flow in the first turn is counter-clockwise
        SKA(sk_out[0])<-SKA(sk_in[0]);       // Convey the system flag
        generate i (1, `NUM, 1) begin
            dummy1 = 1- exp(-(2*i-1)*(2*i-1)*pi*delta_t_0*sp_D_0/(W*W));
            Part1_Sk = - 8*pi*W*W*dummy1/(pow((2*i-1), 4)*pow(pi, 4)*delta_t_0*sp_D_0); // use "-" sign
            ...
        end
        ...
    end

    end
    ...
end
end
endmodule                                    // End of behavioral description
// End of module

```

FIGURE 12. Verilog-A description for a 180° turn involving clockwise flow of the species band. It determines the signs used in Eq. (24), as well as the canceling and strengthening effects of the skew.

The resulting nodal voltages and branch currents are in turn used to calculate the electric field strength (E) and its direction within each element, as well as flow and splitting ratios at intersections (for mixers). With these results and user-provided sample properties (D and μ), the sample speed is then given by $u = \mu E$. Next, values of biofluidic pins at the outlet(s) of each element (e.g., concentration coefficients for micromixers; arrival time, variance, skew and amplitude for separation micro-

chips) are determined. The process starts from the most upstream element, typically the sample reservoirs in mixers and the injector in separation chips in terms of the directional signal flow. As such, both electric and fluidic information in the entire system are obtained.

The mixer operates in steady-state, while transient evolution is critical in separation channels. Transient analysis can be also conducted for separation chips that involve the species band's motion and broadening. An electropherogram (average concentration c_m vs. time) can be obtained at the detector, yielding an intuitive picture of separation resolution between species bands. The transient analysis first calculates for the DC operating points of the amplitude A_{det} , separation time t_{det} and variance σ_{det}^2 of the species band at the detector as described above. Based on these points, the actual read-out time is scanned and the average concentration output c_m is calculated. Assuming the species band does not appreciably spread out as it passes through the detector, c_m is given by

$$c_m = A_{det} \cdot e^{\frac{-(E\mu)^2(t-t_{det})^2}{2(\sigma_{det}^2 + \Delta\sigma^2)}} \quad (28)$$

where t is the actual read-out time and $\Delta\sigma^2$ is the variance growth associated with detection and given in Eq.(27).

Simulation Results and Discussion. In this section, simulation examples of complex EK passive mixers and electrophoretic separation microchips will be presented to verify the behavioral models for biofluidic elements and validate the modeling and simulation methodology. Schematic simulations for micromixers are shown in Fig. 13 and Fig. 14 and Table 2, and those for electrophoretic separation systems are given in Fig. 16-Fig. 21.

Electrokinetic micromixers and mixing networks. Electrokinetic focusing [64], which first appeared as an important fluid manipulation technique in EK LoC systems, also can be applied to speed up mixing, especially in reaction kinetics studies [65]. Fig. 13 illustrates an EK focusing mixer and its system-level schematic. In the discussion below, subscripts i , s and o respectively denote the middle-input, side and output mixing channels. The cross intersection where sample a (red) from the input channel is focused by buffer or sample b (blue) from both side channels, is modeled as two concatenated converging intersections. The flow ratio (the ratio of the flow rate of the middle-input stream to the total flow rate) of sample a is $s = I_i / (2I_s + I_i)$.

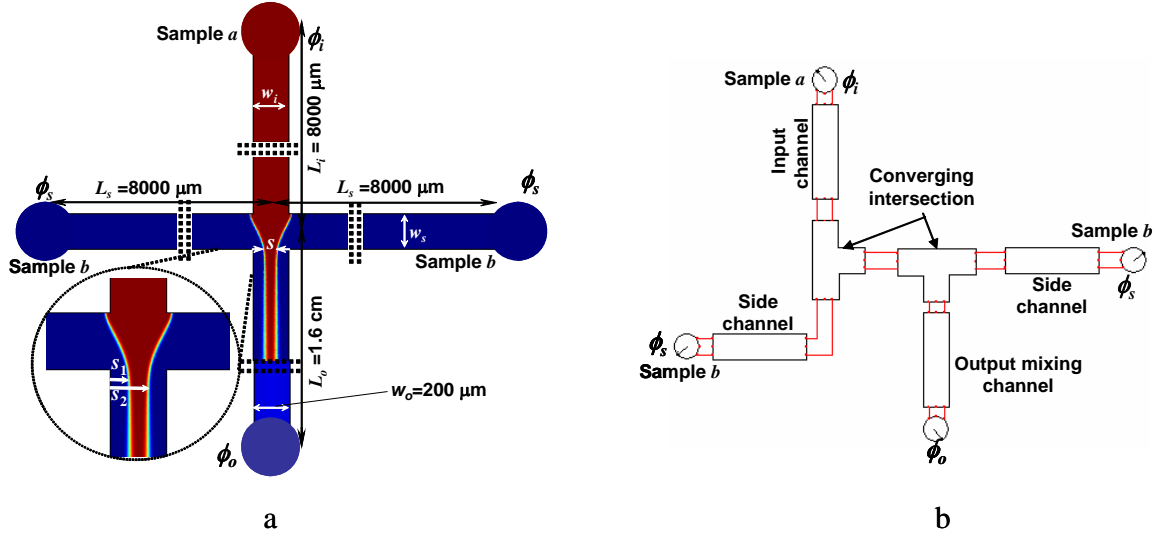


FIGURE 13. (a) An electrokinetic focusing micromixer. Sample *a*, flowing from the top input channel to the intersection, is pinched by sample *b* (or buffer) from both side channels. Then samples mix in the bottom mixing channel. (b) Its hierarchical schematic representation. The triple input and one output cross intersection is modeled as a cascade connection of two converging intersections.

Fig. 14(a) shows numerical and schematic simulation results of sample *a* concentration profiles at the mixing channel outlet for two flow ratios $s = 0.1$ and $s = 1/3$. During simulations, reservoir voltages (ϕ_i and ϕ_s) are chosen to vary s while keeping E (143 V/cm) and the sample residence time fixed in the mixing channel. Excellent agreement between numerical and schematic simulation results is found with the worst-case relative error of 3% at $s = 0.1$. The results are also compared with those from a T-type mixer that has the same electrical field (in the mixing channel), channel

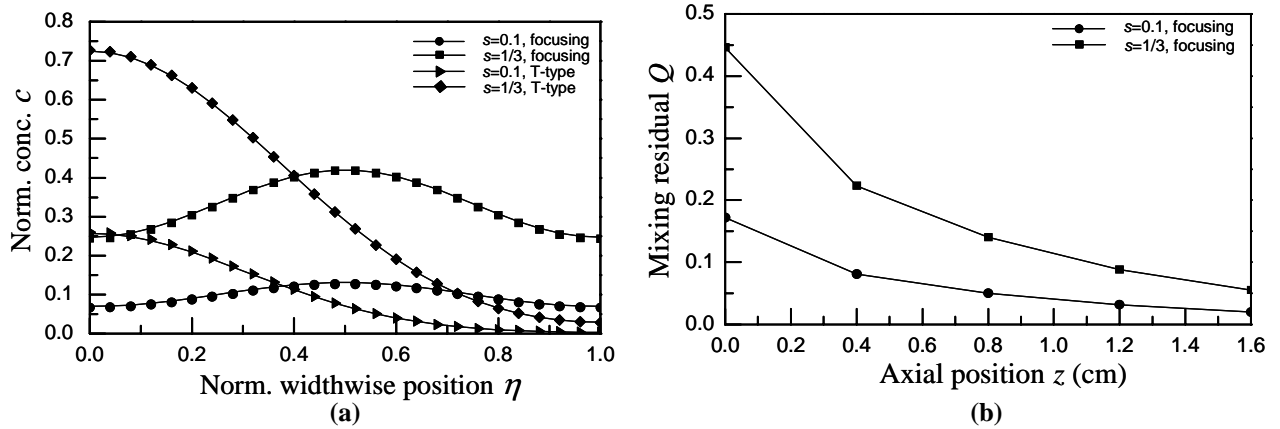


FIGURE 14. (a) Schematic simulation results (lines) compared with numerical data (symbols) on concentration profiles c (sample *a*) along the normalized channel width η for the electrokinetic focusing and T-type mixers. In contrast to the T-type mixer, the focusing mixer considerably improves sample homogeneity due to the reduced diffusion distance between samples; (b) Schematic simulation results on variation of mixing residual Q along axial channel length (data points are connected by lines to guide the eye) for the electrokinetic focusing mixer involving different stream width s . A smaller stream width (e.g., $s = 0.1$) yields a lower initial mixing residual and a more uniform concentration profile at the end of the mixing channel.

Table 2. Comparison of schematic simulation results (sche) with numerical (num) and experimental (exp) data on sample concentrations in analysis channels of serial and parallel mixing networks [60]

| Serial Mixing Network | | | | | | Parallel Mixing Network | | | |
|-----------------------|------------|-----------|----------------|------------|-----------|-------------------------|------------|-----------|-----------|
| Complete Mixing | | | Partial Mixing | | | Complete Mixing | | | |
| channel | c (sche) | c (exp) | c (num) | c (sche) | c (num) | channel | c (sche) | c (exp) | c (num) |
| A ₁ | 1 | 1 | 1 | 1 | 1 | A ₁ | 0 | 0 | 0 |
| A ₂ | 0.37 | 0.36 | 0.378 | 0.48 | 0.496 | A ₂ | 0.83 | 0.84 | 0.832 |
| A ₃ | 0.22 | 0.21 | 0.224 | 0.187 | 0.187 | A ₃ | 0.68 | 0.67 | 0.674 |
| A ₄ | 0.125 | 0.13 | 0.133 | 0.081 | 0.0815 | A ₄ | 0.52 | 0.51 | 0.523 |
| A ₅ | 0.052 | 0.059 | 0.0628 | 0.029 | 0.0315 | A ₅ | 0.35 | 0.36 | 0.354 |
| | | | | | | A ₆ | 0.17 | 0.19 | 0.168 |
| | | | | | | A ₇ | 1 | 1 | 1 |

length and width as the focusing mixer. The focusing mixer considerably improves sample homogeneity, which can be attributed to the reduced diffusion distance between samples. That is, the axial centerline of the mixing channel in the focusing-mixer can be treated as an impermeable wall due to symmetry, hence the diffusion distance is only one-half of that of the T-type mixer. Also, a smaller stream width (e.g., $s = 0.1$) yields a more uniform concentration profile at the end of the mixing channel.

To gain the insight of the influence of the stream width on mixing performance, an index of mixing residual, $Q = \int_0^1 |c(\eta) - c_{avg}| d\eta$, is introduced in Fig. 14(b) to characterize the non-uniformity of concentration profiles, where $c(\eta)$ and c_{avg} are the normalized concentration profile and width-averaged concentration respectively at the detection spot. At the channel inlet ($z = 0$), mixing residual Q strongly depends on s . Asymmetric incoming streams yield a lower Q value (e.g., $Q = 0.18$ at $s = 0.1$ compared with $Q = 0.44$ at $s = 1/3$) and a more uniform initial profile. Along the channel, Q initially drops rapidly and then becomes saturated because the improved sample mixing reduces the concentration gradient and the driving force for further mixing. Thus, a tradeoff between Q and mixer size can be captured by the behavioral models presented in this paper to achieve designs of both effective and efficient micromixers.

These parameterized and reusable behavioral models are well suited to study complex mixing networks [60], in which an array of sample concentrations can be obtained at multiple analysis channels by geometrically duplicating units with a single constant voltage applied at all reservoirs.

Table 2 shows the comparison of schematic simulation results with experimental and numerical data on sample (rhodamine B) concentrations in analysis channels A₁–A₅ in the serial

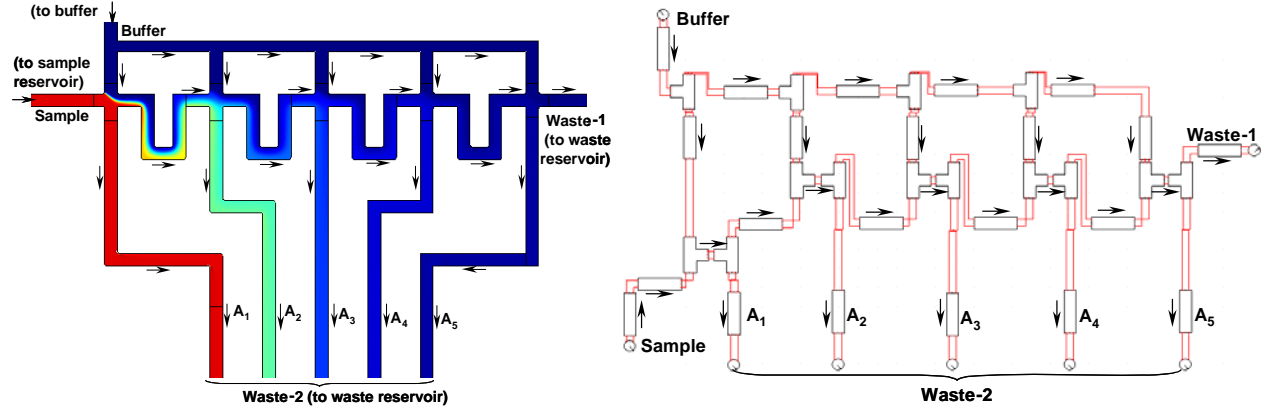


FIGURE 15. An electrokinetic serial mixing network [60] and its hierarchical schematic representation. The network consists of reservoirs, mixing channels, T- and cross-intersections. Sample and buffer is released and collected by the reservoirs. In the composable approach, the serial mixing network is represented as a collection of interconnected mixing elements composed of microchannels, converging intersections and diverging intersections.

mixing network (Fig. 15). Both complete and partial mixing cases are investigated. When a voltage of 0.4 kV is applied at the sample and buffer reservoirs with the waste reservoirs grounded, sample mixing in channels S_2 – S_5 is width-wisely complete. Excellent agreement among the schematic simulation, numerical analysis and experimental results is found (with an average error smaller than 6% relative to experiments). In contrast to the electric resistor-based models [22][32][60] that take advantage of the analogy between EK flow and electric current and require post-calculations of concentrations from current distributions in the network, our behavioral models directly yield the concentration value in each analysis channel. In addition to complete mixing, partial mixing case is also schematically simulated. A voltage of 1.6 kV, as used in the experiments in [60], is applied at the sample and buffer reservoirs with the waste grounded, which increases the EK velocity and then decreases the residence time of the sample by four folds in channels S_2 – S_5 . Thereby, the mixing in channels S_2 – S_5 is width-wisely incomplete, and the amount of sample shunted to channels A_1 – A_5 depends on not only the electric currents in the network but also the sample concentration profiles at the exits of channels S_2 – S_5 , which violates the assumption for the analogy between EK flow and electric currents and hence the resistor-based modeling becomes invalid. However, it can be readily simulated by our behavioral models. In the schematic, the cross-intersection is modeled as a combination of the converging and diverging intersections, in which the sample concentration profiles of the incoming and outgoing streams are accurately captured. Results from schematic simulations are compared with numerical data in Table 2 (a comparison to experimental data is not

allowed due to a lack of knowledge of sample properties. Hence a diffusivity of $D=3\times10^{-10}$ m²/s and an EK mobility of $\mu=2.0\times10^{-8}$ m²/Vs are assumed in numerical simulations). Very good agreement can be observed with an average error of 4%. At the cross-intersection following channel S₂, the amount of sample shunted to A₂ is more than that predicted by the complete-mixing case due to the non-uniform sample profiles at the intersection's inlet. Consequently, concentrations in channels A₃–A₅ show the lower values, which agrees with experimental observations [60]. Netlisting and schematic simulation of this example take 20 seconds on a multi-user, 2-CPU 1-GHz Sun Fire 280 processors with 4 GB RAM for the first time simulation, and less than a second for subsequent iterations, leading to a 1000–20,000× speedup.

In addition to the serial mixing network, the parallel mixing network [60] can be hierarchically represented and simulated in a similar fashion and excellent agreement among schematic simulations results, numerical analysis and experimental data (with an average error of 3.6% relative to experiments) is also found [59].

Electrophoretic separation microchips. Schematic simulation results for electrophoretic separation microchips are shown in Fig. 16–Fig. 21. In Fig. 16 and Fig. 17, a serpentine electrophoresis column of two complementary turns is used to separate an analyte comprised of two species *a* ($D=3.12\times10^{-10}$ m²/s, $\mu=1.2\times10^{-8}$ m²/sV) and *b* ($D=2.72\times10^{-10}$ m²/s, $\mu=1.1\times10^{-8}$ m²/sV) with $E=600$ V/cm. Experimental data [66] on variance vs. time of species *a* are compared with DC schematic simulations in Fig. 16, showing excellent agreement with the worst-case relative error of only 5%. Again, netlisting and DC simulation for this example take 20 seconds for the first time and less than a second for subsequent iterations, leading to a 500–10,000× speedup (higher speedup can be obtained for a more complex chip topology or a less diffusive species as shown in Fig. 20). The first turn skews the species band and accordingly incurs abrupt increase in variance. During the species band's migration in the long inter-turn straight channel, the transverse diffusion smears out most of the skew and presents a nearly uniform band before the second turn. The second turn then distorts the band again in the opposite direction, leading to another turn-induced variance that is equal to the one caused by the first turn. Fig. 17 shows electropherograms of both species from three detectors. The spacing between concentration peaks of species *a* and *b* increases as they migrate through channels, but due to the band-broadening effect, the amplitude decreases

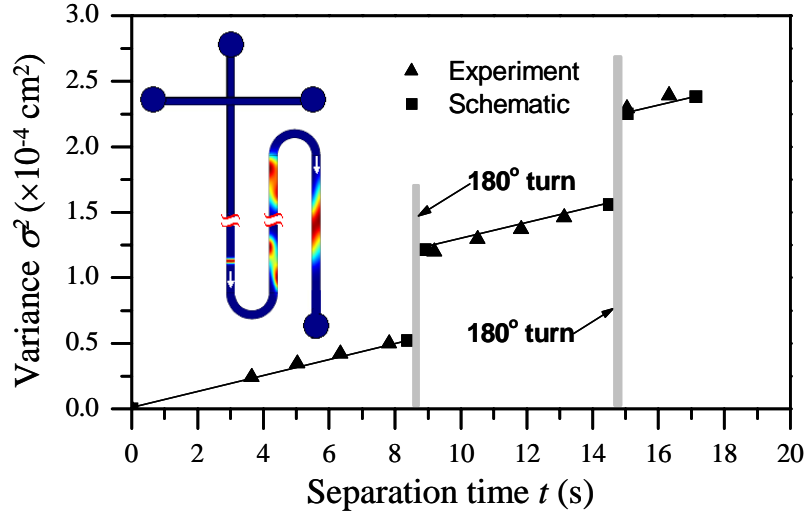


FIGURE 16. Comparison of experimental data [66] with DC schematic simulation on variance σ^2 vs. separation time t of species a in a serpentine electrophoretic separation microchip of two complementary turns. The grey bars represent the residence time of the sample within the turns. The first turn skews the species band and accordingly incurs abrupt increase in variance. The transverse diffusion in the inter-turn straight channel smears out most of the skew and presents a nearly uniform band before the second turn (see the inset of numerical simulation plot). The second turn then distorts the band again in the opposite direction, leading to another turn-induced variance.

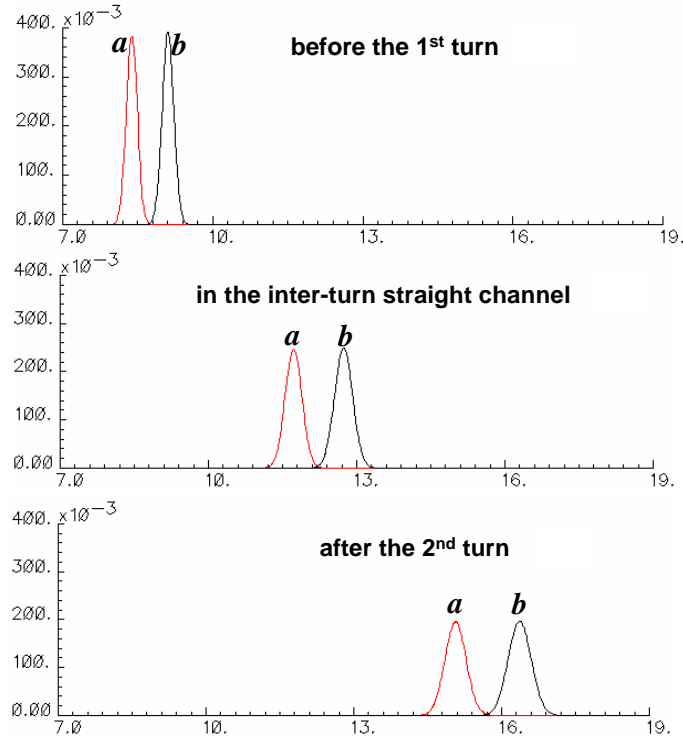


FIGURE 17. Transient analysis simulates the electropherograms output from three detectors, which are respectively arranged before the first turn (top trace), in the middle of the inter-turn straight channel (middle trace) and after the second turn (bottom trace). Attributed to the difference in electrokinetic mobility, the spacing between concentration peaks of species a and b increases as they migrate through channels. The dispersion effect leads to the continuous decreases in the band amplitude.

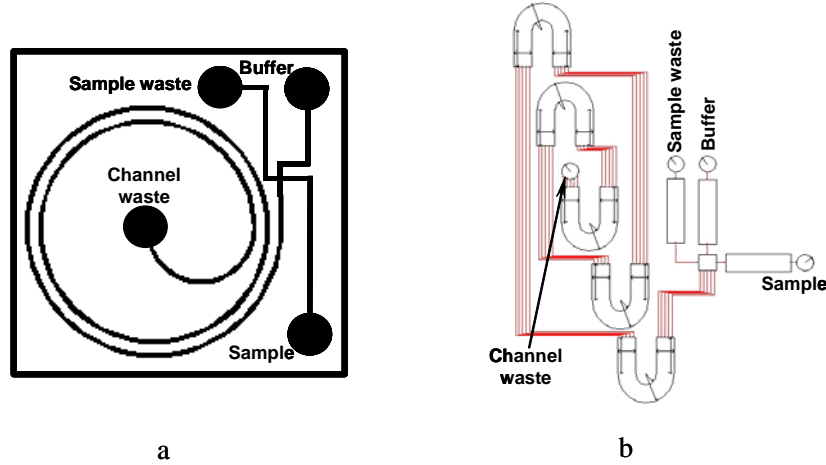


FIGURE 18. (a) A spiral electrophoretic separation microchip [20]. It consists of five turns with continuously decreased radius (1.9, 1.8, 1.7, 1.6 and 0.8 cm). Within them, species Dichlorofluorescein flows in the same direction (clockwise). (b) Its hierarchical schematic representation.

consecutively.

In Fig. 18, the dispersion of Dichlorofluorescein in a complex spiral separation microchip of five turns is simulated and compared with experimental results [20]. Spiral channels differ from the serpentine in that the skew and variance always increase with the turn number, as the band skew in all turns has the same sense and does not cancel. A scalar index of plate number N_s to characterize the resolving power of the electrophoresis chip is defined $N_s = L_{tot}^2 / \sigma^2$, where L_{tot} is the total separation length from injector to the detector. The higher the plate number, the better separation performance achieved by the chip. The linear growth of the plate number with electric field implies that molecular diffusion is the major dispersion source in such a system Fig. 19), as molecular diffusion decreases linearly as electric field increases. The worst-case relative error of 12% is considered acceptably small considering the uncertainties in the knowledge of species diffusivity [20].

Fig. 20 illustrates a hybrid electrophoretic separation microchip [67] and its schematic representation including both spiral and serpentine channels. Due to the difficulty of accounting for the coexisting skew canceling and strengthening effects in such a topology, it has not been effectively investigated since it was proposed [67]. Fig. 21 shows schematic simulation result on the variance of a species band vs. time in such a chip, as well as its comparison with numerical data. A low species diffusivity of $D = 1 \times 10^{-11} \text{ m}^2/\text{s}$ is chosen to analyze the highly convective dispersion

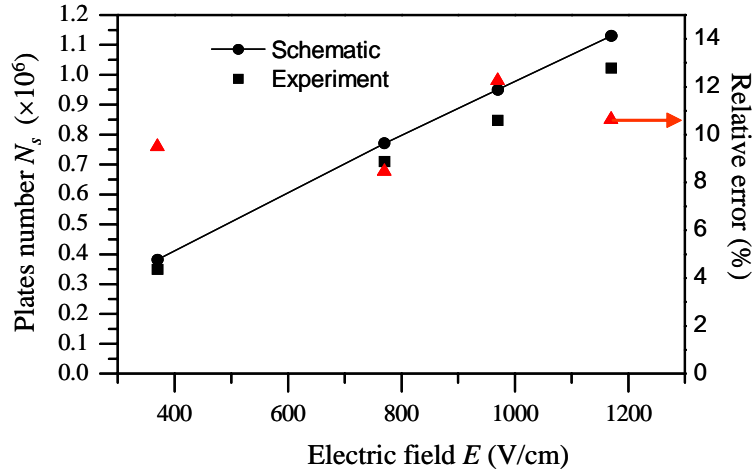


FIGURE 19. Comparison of schematic simulation results to the experimental data on theoretical plate number N_s vs. electric field E . Right axis shows the relative error between simulation and experiments. The linear growth of the plate number N_s with electric field E implies that molecular diffusion is the major dispersion source in such a system.

that has not been considered by the previous example in Fig. 16 (other properties and parameters are the same as those of sample *a* in Fig. 16). Highly convective dispersion is practically important for microchip electrophoresis of the species with low diffusivity, such as the separation of DNA in a

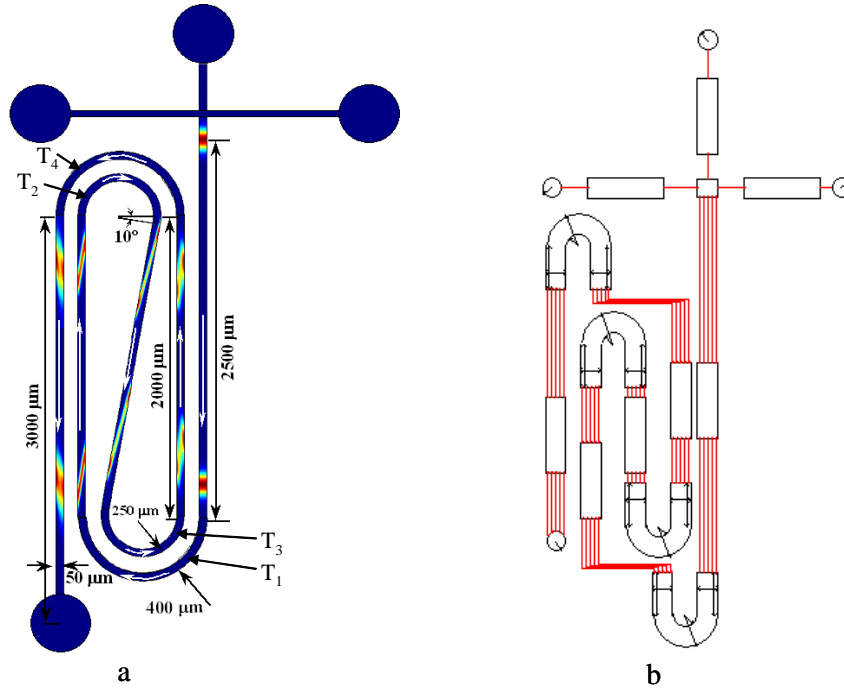


FIGURE 20. (a) A hybrid electrophoretic separation microchip. It consists of both spiral and serpentine channels. Species flows in the clockwise direction in both turns T_1 and T_2 (spiral topology), thereby T_2 strengthens the sharp skew generated by T_1 . The skew almost persists through the inter-turn straight channel between T_2 and T_3 and is significantly cancelled out by T_3 (serpentine topology). (b) Its hierarchical schematic representation.

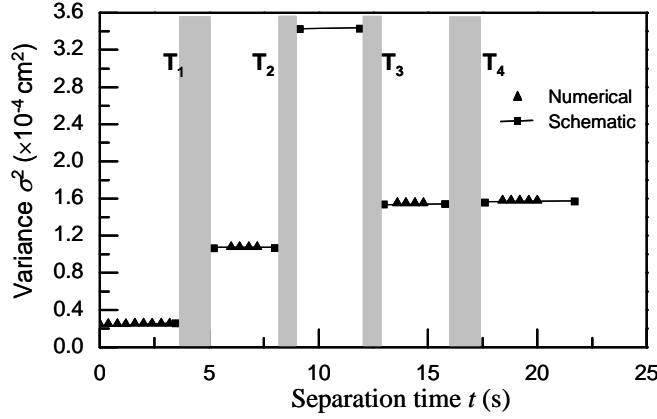


FIGURE 21. Comparison of numerical data with DC schematic simulation on variance vs. separation time in the hybrid electrophoretic separation microchip. Very sharp skew (see Fig. 20) is generated and the variance accumulates after turns T_1 and T_2 due to their spiral topology. The skew almost persists through the inter-turn straight channel between T_2 and T_3 and is significantly cancelled out by T_3 attributed to their serpentine topology, which as a result yields a drastic variance drop after T_3 .

gel or sieving matrix [66][68]. It is shown in Fig. 20 that since species flows in the clockwise direction in both turns T_1 and T_2 (spiral topology), T_2 strengthens the sharp skew generated by T_1 , leading to a more skewed band and a higher variance. Due to the small species diffusivity, the skew almost persists through the inter-turn straight channel between T_2 and T_3 and is significantly cancelled out by T_3 , which as a result yields a drastic variance drop in T_3 (serpentine topology). However, the skewed band after T_3 is overly corrected by T_4 and a counter-skew is shown afterward. Excellent agreement between the schematic and numerical simulation results with 1% relative error and tremendous speedup up to 400,000 \times have been achieved in Fig. 21. This is the first time that the highly convective dispersion in the hybrid electrophoresis microchip at this complexity level has been accurately and efficiently simulated by analytical models.

4.1.5. Injector Models

Injector Basic Operation and Topologies. The injector prepares the sample from the synthesis stage of the LoC for measurement in the separation stage. This is done very simply by the intersection of one or more channels as shown in Fig. 22. It should be noted that injectors are classified not only by topology, but also by the field arrangements used to shape the fluids inside--for example the cross and gated-cross. The earliest on-chip injectors were the simple tee [69][70]. The tee operates in two stages by first flowing analyte past the empty separation channel. Once enough material has passed the separation channel, the flow is reversed by modifying the potential on the separation channel to create an electric field, capturing a plug of material. This method has poor control over

the size of the material plug, [71], and has mostly be replaced by the cross, [72], and double-tee, [73]. The cross and double-tee also operate in two stages and they operate on similar principles; the double-tee simply provides a larger injected band due to the channel offset. In both injectors, material is first moved into the channel intersection, called the injection chamber. While material is moving into the chamber, electric fields from the side channels (accessory fields) constrict the flow to provide a thinner band to the separation channel. In the second stage, the fields change direction to flush the thin column of material into the separation channel. In the second stage, the fields change direction to flush the thin column of material into the separation channel. Simultaneously, the side channels (now the top and bottom channels) have fields directed away from the injection chamber to prevent additional material from flooding into the separation channel, guaranteeing a small clean plug [62][74][75]. The gated-cross has the same topology as the regular cross, but operates with significantly different electric fields, [62][76][77][78][79]. The gated-cross is desirable when repeated injections must be performed in rapid succession. The injector operates by running the material at a 90 degree angle from the input channel using an opposing field to form a gate. The gate is released for a given time to obtain a plug of the desired size and then closed. When the gate is closed the opposing field flushes the plug into the separation channel. Meanwhile the original material continues to flow on the other side of the gate and is immediately ready for another injection. The double-cross injector is similar to the regular cross, but includes an extra set of accessory channels to create

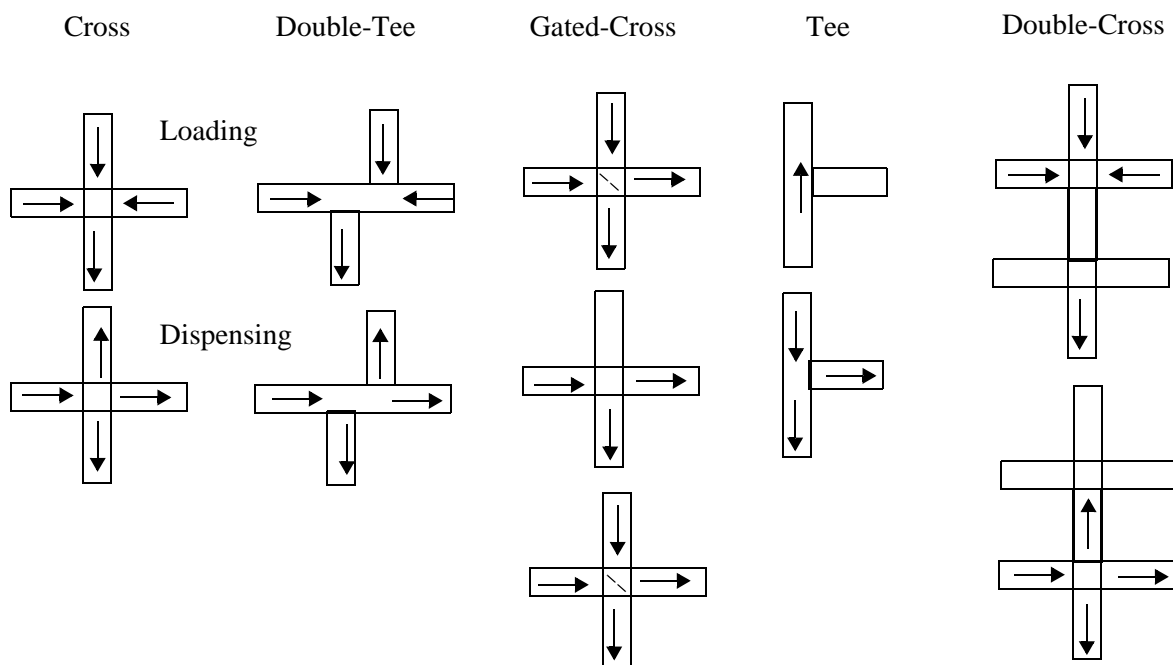


FIGURE 22. Basic operating stages for various injector topologies. The most common topologies are the cross, double-tee, and gated cross. The earliest on-chip injectors were the tee.

an extra thin and uniform stream for plug extraction, [80]. The additional complexity of controlling six channels simultaneously is typically not worth the incremental improvement over the regular cross, and thus this topology has not seen wide acceptance. Other lesser-used topologies and control methods have been explored for specific applications such as the multi-legged fork [81], which provides discrete volumes based on the number of legs fabricated and the distance of the inflow and outflow legs, and the gated-double-tee and triple-tee [82], which provide modifications to the regular double-tee.

The goal of all of these injection methods is to increase the efficiency of the separation channel. The injector should provide a small well-defined plug of material with maximum concentration for ease of separation. The smaller the plug is, the shorter the distance each species' peak center must travel to be resolvable, thus shorter channels can be used, allowing room for more complex system integrations. An example of this is efficiency shown in Fig. 23. The parameters of the simulation are shown in the caption of the figure, and the results show that a wider band requires a longer separation channel in order to get the same resolution.

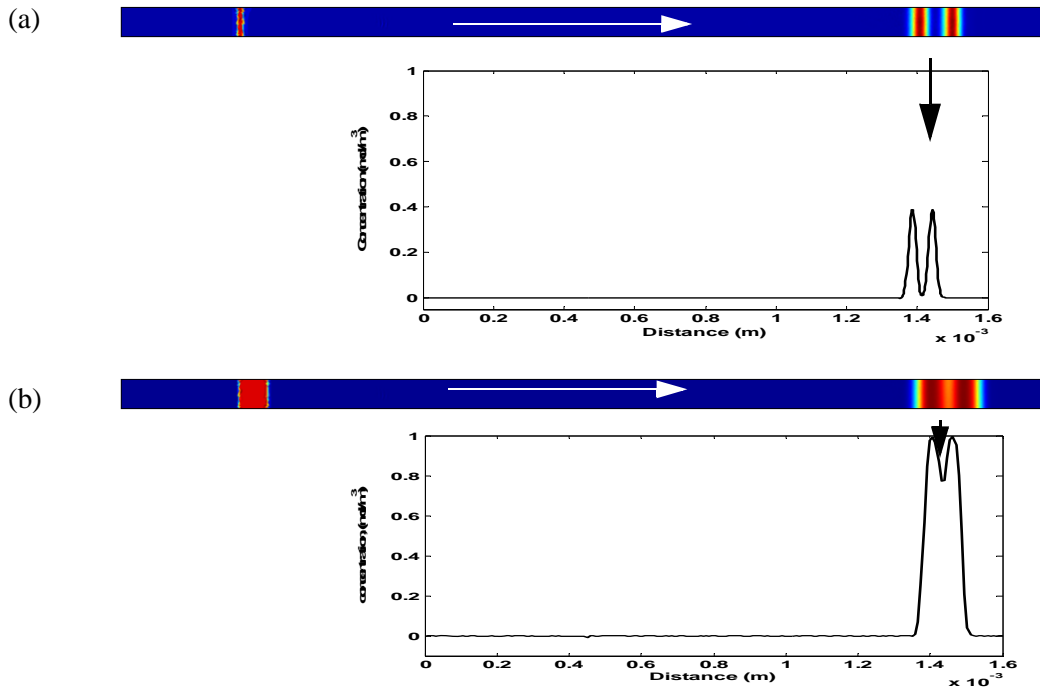


FIGURE 23. Two species are combined and a band of material representative of an injector formed plug is placed at the beginning of the channel. Species one and two have a diffusivity of $1 \times 10^{-10} \text{ m}^2/\text{s}$. Species one has a mobility of $5.25 \times 10^{-7} \text{ m}^2/\text{Vs}$, whereas species two has a mobility of $5.5 \times 10^{-7} \text{ m}^2/\text{Vs}$. After traveling in a 50000 V/m electric field for 1.2 mm , the separation of the two species bands is measured. In the first example (a), the plug has a width of $10 \mu\text{m}$. In the second example (b), the plug has a width of $50 \mu\text{m}$. Even though the plug in (b) contains more material, it is not as easy to resolve as the smaller plug in (a).

Injector Modeling Parameters. The parameters can be categorized into one of three categories: (1) Those describing the fluidic properties of the analyte, (2) those describing the applied electric fields, and (3) those describing the geometry of the given injector's topology. The fluidic properties include the physicochemical attributes of the buffer fluid and the transported chemical species such as electrokinetic mobilities and diffusivities. The applied electric fields provide the stimulus and driving forces for the fluids. These fields are characterized by their direction and magnitude in each of the channels. In general, the conservation of current and ohmic conduction are inherent to the problem so although there may be N channels, there are only $N-1$ independent field quantities. For each stage of operation the field directions or magnitudes may vary, so for M stages of operation there are $M(N-1)$ independent field parameters. The relevant geometric parameters for injectors are the channel widths and lengths, where transport in the depth direction is typically ignored. The widths of the injector channels near their intersection point are almost always the same, so knowing one width provides information about all channel widths. The effects of unequal channel widths at the intersection has been explored [83]. Unequal channel widths has the ability to enable injections with fewer controlling electrodes, but is not commonly implemented due in part to the difficulty of filling such channels with buffer when dry. The unequal channel resistances make the pressure driven filling more difficult.

Finally, unless noted otherwise the injector is treated as a steady state device. Each stage of operation is run until stable equilibrium is reached. For the cross and double-tee loading stages, this equilibrium is achieved in all cases, except when the side channels are left floating (no accessory electric fields), permitting diffusion into the side channels that floods the chip. In practice, the injector is only loaded for a limited time to prevent flooding the side channel with diffusion in the absence of accessory fields. In general, this time is long enough to reach a stable steady state for the instances when there are accessory electric fields in the side channels. In this work the following time is used:

$$t_L = 8.4 \left(\frac{L_1}{U_1} + \frac{5\lambda}{U_{12av}} + \frac{L_3}{U_3} \right) \quad (29)$$

where L_I is the length traveled from the loading stage input and waste port to the channel intersection, U_I is the velocity in the input channel, λ is the offset length of the channels--zero for the cross, U_{12av} is the average velocity of the input channel and the waste channel, L_3 is the length from the channel intersection to the waste port, and U_3 is the velocity in the waste channel. The factors of 8.4

and 5 are empirically chosen to ensure sufficient time for steady state to be achieved when accessory fields are implemented.

The loading stage is characterized by a time that allows the band to completely exit the injection chamber:

$$t_d = \frac{L_d}{U_d} \quad (30)$$

where L_d is a characteristic length for the band to travel to exit the injection chamber and U_d is the velocity of the band in the separation channel. Beyond this point, the effects of the separation channel determine the dispersion of the band.

Injector Leakage. For proper operation or simulation all of the parameters and their appropriate *ranges* must also be defined. This defines the design space for the injector subsystem. An injector is only properly operating for certain values of the electric field, and the configuration of electric fields contributes to the classification of an injector. For example, if the input electric field reverses direction and prevents the injection chamber from filling with chemical species then the device will not have the material necessary to form a plug for separation. Another important implication is that the fields must be applied in the proper ratio for the dispensing stage to prevent flooding the separation channel with material from the upstream mixer or reservoir. This is achieved by requiring the field in side channels during dispensing to be larger than some multiple, α , of the driving field in the separation channel. This is depicted in the cartoon of Fig. 24, and the factor, α , depends on the

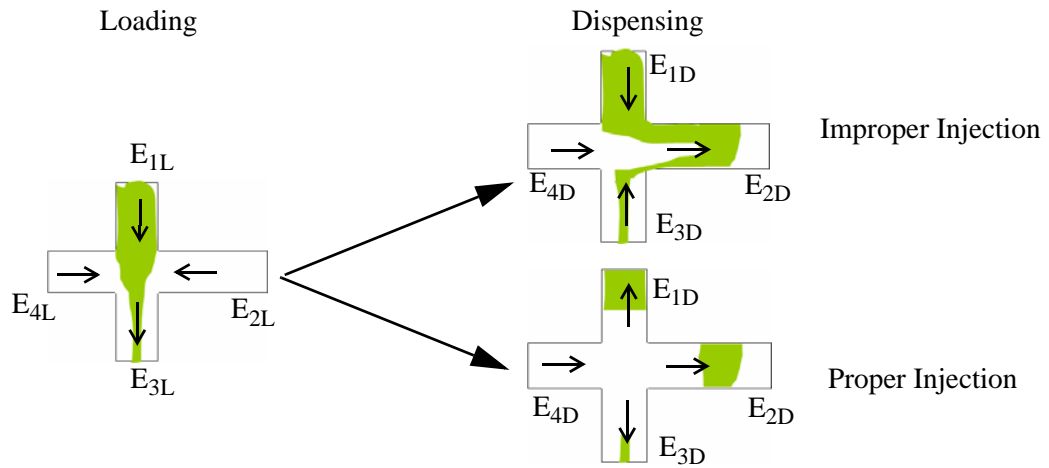


FIGURE 24. Proper operation of an injector requires the identification of control parameters and their applicable ranges. Here identifying the fields (arrows) is the first step, but then the magnitude and ratios must be additionally constrained for proper operation ($E_{1L} > 0$, $E_{1D}/E_{2D} > \alpha$, where $E_{1D} = E_{3D}$)

Peclet number for this dispensing stage, since more diffusion will be conducive to more leakage. Applying the Buckingham-Pi theorem identifies the essential non-dimensional parameters to be a field ratio in the loading stage (L) and dispensing stage (D), $\varepsilon_L = E_{2L}/E_{3L}$, $\varepsilon_D = E_{1D}/E_{2D}$, and the Peclet numbers in the loading and dispensing stage, $Pe_L = U_1 w / \kappa$, $Pe_D = U_2 w / \kappa$. Where κ is the diffusivity, U_i is the velocity in channel i , and w is the channel width.

To measure α , simulations were done for a cross injector. To quantify whether there was significant leakage, the concentration was measured at the immediate edge of the injection chamber adjacent to the separation channel, after the band had traveled six channel widths downstream. If the ratio of the peak of the injected band to the value at the edge of the injection chamber, exceeds the ratio of the peak of a Gaussian distribution to its value at 3σ , leakage is defined as excessive. The following relation defines the region of acceptable leakage operation:

$$\frac{C_{max}}{C_{min}} \geq \exp\left(\frac{9}{2}\right), \quad (31)$$

where C_{max} is the peak of the injected band and C_{min} is the concentration at the exit of the injection chamber.

The leakage obviously depends on the ratio of the accessory fields in the side channels to the driving field in the separation channel and the Peclet number for the dispensing stage. To test for dependence on the loading stage parameters, the field ratios and Peclet number for the loading stage were varied as well. The results are displayed in Fig. 25 and show a slight dependence on the loading stage field ratios. The more constricted the band is during the loading stage, the less leakage is likely in the dispensing stage. The worst case (low loading field ratios and low loading Peclet number) is used to create a bound to restrict the design space of the cross and double-tee injectors:

$$\varepsilon_D > \frac{1}{-16.65} \ln\left(\frac{Pe_D}{8490}\right) \quad (32)$$

Injector Output Plug Gaussian Representation. The size and shape of the plug of material injected into the separation channel is the modeled output of the injector. The numerical results simulated at all of the points defined by the injector design space contain all of the information regarding the plug. To efficiently represent the plug, the results of the numerical simulation are processed to extract the equivalent Gaussian peak-height and variance. The injector model has the following form:

$$\dot{f}(\bar{\pi}) = \begin{bmatrix} \sigma^2/w^2 \\ C_p/C_o \end{bmatrix} \quad (33)$$

where $\bar{\pi}$ are the input parameters for the specific injector device being modeled, σ^2 is the variance of the widthwise average concentration from the numerically simulated injector, w^2 is the channel width normalizing factor, C_o is the normalizing concentration -- typically the output concentration of the upstream mixer, and C_p is the peak-height of an equivalent Gaussian for the transversely averaged concentration of the numerically simulated injector such that the effective Gaussian and the actual plug have the same mass,

$$C_p = \frac{m}{\sqrt{2\pi\sigma^2}} \quad (34)$$

where m is the mass of the band, and σ^2 is the band variance.

This is an efficient representation of the band since the effects of diffusion will always act asymptotically to make any band Gaussian. However, since diffusion takes time to homogenize the band, there is a limit on the applicability of this method. This bound can be identified based on several time constants associated with the downstream separation channel. The effective Gaussian representation will be an accurate representation as long as the detector is far enough downstream to

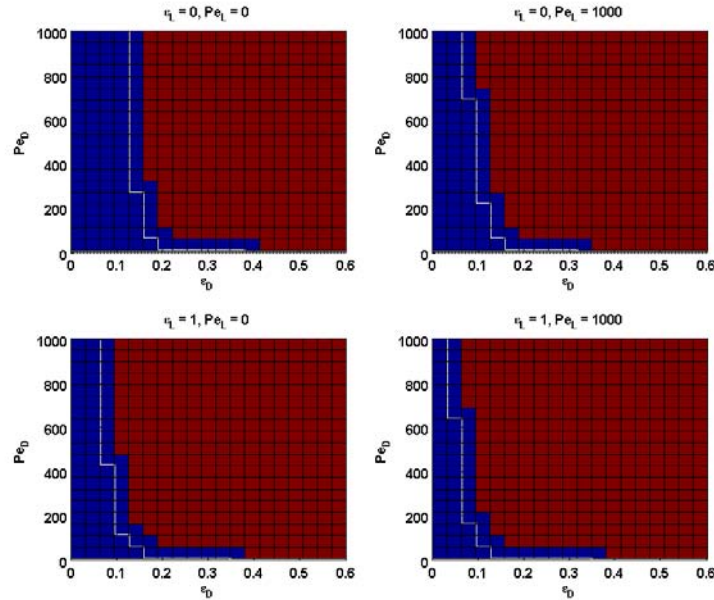


FIGURE 25. Leakage results for the cross and double-tee injectors. The values of the loading stage parameters are fixed, but different in each of the four plots. The blue region defines infeasible operation, and the red region defines feasible operation for the dispensing stage field ratios, ε_D and Peclet number Pe_D .

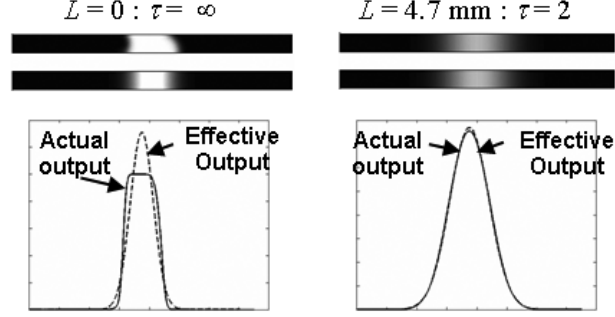


FIGURE 26. Comparison of actual injection plug to its effective Gaussian representation. The actual output of a double-tee injector is in the top channels, the effective Gaussian model is shown in the bottom channels. The band on the right has traveled 4.7mm, which is when $\tau = 2$. For these simulation $\pi_{1D} = 186$, $\pi_{2D} = 186$, $\pi_{3D} = 1/8$, and $\pi_{4D} = 0.57$, $\pi_{5D} = 2$ (definitions can be found in Table 3).

allow diffusion to work and before there are any additional sources of dispersion such as turns in the channel. In other words, the time it takes particles to diffuse across the width of the channel, w ,

$$t_{\kappa} = \frac{w^2}{2\kappa}, \quad (35)$$

must be shorter than the time it takes particles to reach the first channel bend or detector, L ,

$$t_c = \frac{L}{U}, \quad (36)$$

Using these times and requiring $t_c > t_{\kappa}$ the following bound can be used to ensure that the band is accurately represented by a Gaussian before being detected at the end of the separation channel:

$$\tau = \frac{Pe_w}{L/w} < 2 \quad (37)$$

where Pe_w is the Peclet number based on the channel width. One implication of this bound is that it not accurate to query the detailed shape of the plug immediately after the injector, since the actual plug has not had a chance to homogenize into its ultimate Gaussian shape.

As an example of the effectiveness of this representation, Fig. 26 shows the result of a band created from a double-tee injector as it travels down a straight separation channel. Immediately after the injector the differences in the plug are apparent, but as the actual band travels through the channel it becomes Gaussian due to the effects of diffusion and is then indistinguishable from the effective Gaussian plug.

Cross Topology. The standard operation for a cross injector is shown in Fig. 27. It is a two stage

Table 3. Cross Injector Simulation Parameters

| Simulation Parameters | System Non-Dimensional Parameters | Non-Dimensional Parameter Constraints |
|--|---|--|
| <u>Input</u> | <u>Input</u> | <u>Input</u> |
| μ - Electrokinetic mobility | $\pi_{1c} = E_{2L}/E_{1L}$ | $\pi_{3c}, \pi_{4c} \in [10, 5000]$ |
| K - Diffusion coefficient | $\pi_{2c} = E_{1D}/E_{2D}$ | $\pi_{1c}, \pi_{2c} \in [0.01, 5]$ |
| $E_{1L}, E_{2L}, E_{3L}, E_{4L}$ - Loading electric field | $\pi_{3c} = \mu E_{3L} w / K$ | $E_{2L} = E_{4L}$ |
| $E_{1D}, E_{2D}, E_{3D}, E_{4D}$ - Dispensing electric field | $\pi_{4c} = \mu E_{2D} w / K$ | $E_{1D} = E_{3D}$ |
| w - Channel width | | $\pi_{2c} > 1/16.65 \log(8490/\pi_{4c})$ |
| C_o - Analyte concentration | | |
| <u>Output</u> | <u>Output</u> | |
| C_p - Equivalent Gaussian Peak | $\underline{C}_p = C_p / C_o$ | |
| σ^2 - Equivalent Gaussian Variance | $\underline{\sigma}^2 = \sigma^2 / w^2$ | |

device where the material from the upstream mixer is first loaded into the intersecting cross channel region. This is done using one configuration for the electric fields. Once this region has been filled, the electric fields are modified to drive a plug of fluid into the separation channel. Of the fluidic, electric, and geometric parameters mentioned above, the ones relevant to the cross injector are summarized in Table 3, along with the equivalent non-dimensionalized parameter set. The use of non-dimensional parameters using the Buckingham- π theorem [84], reduces the dimensionality of the

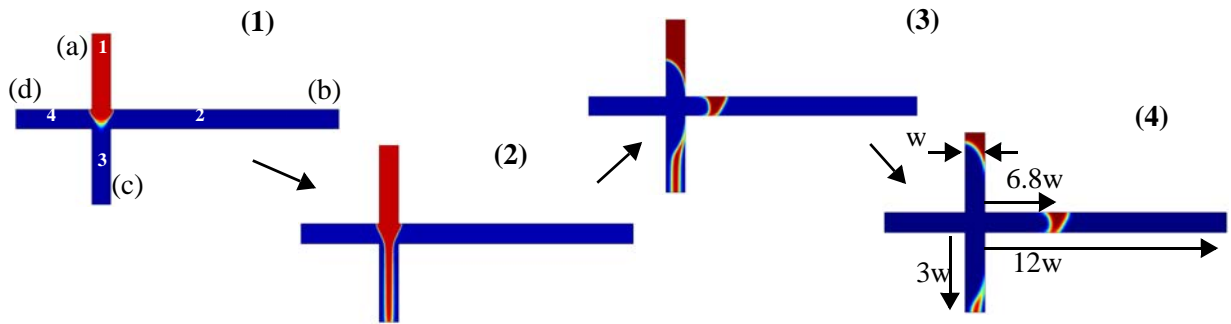


FIGURE 27. Snapshots of the transient injection process for the cross injector using electrokinetic transport. Starting on the left, the analyte in red is drawn from the sample reservoir or upstream mixer at the top (1). After filling the injection chamber (2), the applied potential is changed to inject a plug of analyte while evacuating leftover sample to the north and south to prevent leakage (3,4). The three shorter channels are three channel lengths long and the long channel is 12 channel lengths long.

simulation space from nine independent dimensions (not eleven, since the values for the electric fields are constrained by conservation of current) to four, where the implicit assumption of symmetric shaping ($E_{2L} = E_{4L}$, $E_{1D} = E_{3D}$) has been made.

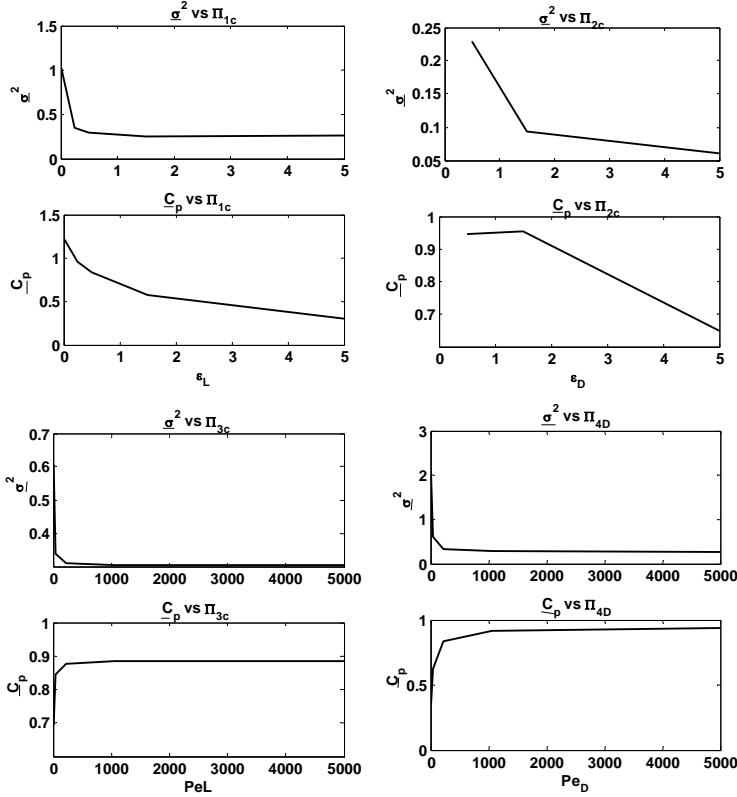
The analyte fluidic property parameters are the mobility, μ (m^2/Vs) and diffusivity, κ (m^2/s). The stimuli parameters are the field strengths, E_{iL} , E_{iD} (V/m), in each injector leg for loading (L) and dispensing (D) stages. The parameter describing the injector's topology is the channel width, w (m), with an implicit assumption that all channels are the same size. The non-dimensional parameters describing the operation are given in the second column; they are the ratio of accessory fields to driving fields in loading and dispensing stages, π_{1c} , π_{2c} and the Peclet numbers in both stages, π_{3c} , π_{4c} .

In order to train a NN model, an appropriate set of model outputs must also be selected. In the case of the injector device the output is the set of parameters that describe an effective Gaussian matching the variance and mass of the plug which injected into the separation channel. The outputs shown in Table 3 are the output peak concentration normalized to the average input species concentration, and the output band variance normalized to the channel width squared.

In order to define the points that will sample the design space defined by the non-dimensional input parameters of Table 3, we first sample the response surface along traces for several nominal values of the input parameters, as shown in Fig. 28a. Several useful pieces of information are seen in these graphs. First, the Peclet scale is exponential, and should be sampled on a logarithmic scale. The field ratio scales are also roughly exponential and can also be sampled on a logarithmic scale. The scales of the non-dimensional parameters are chosen to encompass a large region of the practically realizable injector device operation. The scales are listed in the last column of Table 3. With the information of the design space traces, the entire volume can be sampled using Gaussian sampling. The results of this sampling for various projections, two-parameters at a time, are shown in Fig. 28b. The lower left corner of the π_{4d} vs. π_{3d} plot shows points that were moved out of the infeasible region of the design space. The total of 900 simulations were used to sample the design space.

In order to select the appropriate size for the neural network, the KFCV technique is used to measure the generalization error. A plot is created, shown in Fig. 29, of the cross validation measured error as a function of the number of neurons in the hidden layer. Since the Levenberg-Mar-

(a)



(b)

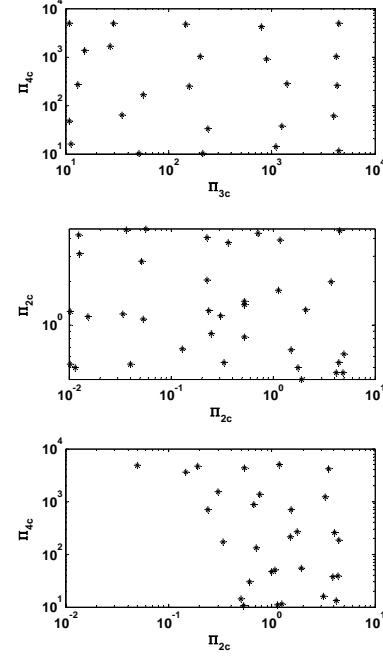


FIGURE 28. (a) Traces through the design space for fixed parameters of $\pi_{1c} = 0.5$, $\pi_{2c} = 0.5$, $\pi_{3c} = 400$, $\pi_{4c} = 400$. (b) Scatter plots of the design space sampled for training the neural network model.

quardt algorithm has a random starting point, each point on the plot is the average of 15 repeated KFCV tests. The calculations for the cross validation and neural network construction are done in MATLAB. Here a NN is created with the topology similar to that in Fig. 6b, except a single layer of hidden neurons is used. The hidden layer neurons use the activation function of Eq. (10) and the output neurons use the linear activation function in Eq. (11). The training algorithm is the Levenberg-Marquardt algorithm [56], for a maximum of 350 iterations. The results of the validation test are shown in Fig. 29. The result shows that there is an increasing error for a low number of hidden nodes due to large biasing error, and increasing error for a high number of hidden nodes due to larger variance errors, with a minimum near 50 nodes for both models. This confirms the intuition that simple networks will have trouble fitting all of the original data points due to lacking degrees of freedom, whereas highly complex networks will hit the original data points easily, but vary widely on data not in the original data set, due to too many degrees of freedom. The noise in the plot of Fig. 29 decreases with an increasing number of averages at the cost of additional computation time. Thus this type of plot can be interpreted to say that on average networks with 50 hidden layer neu-

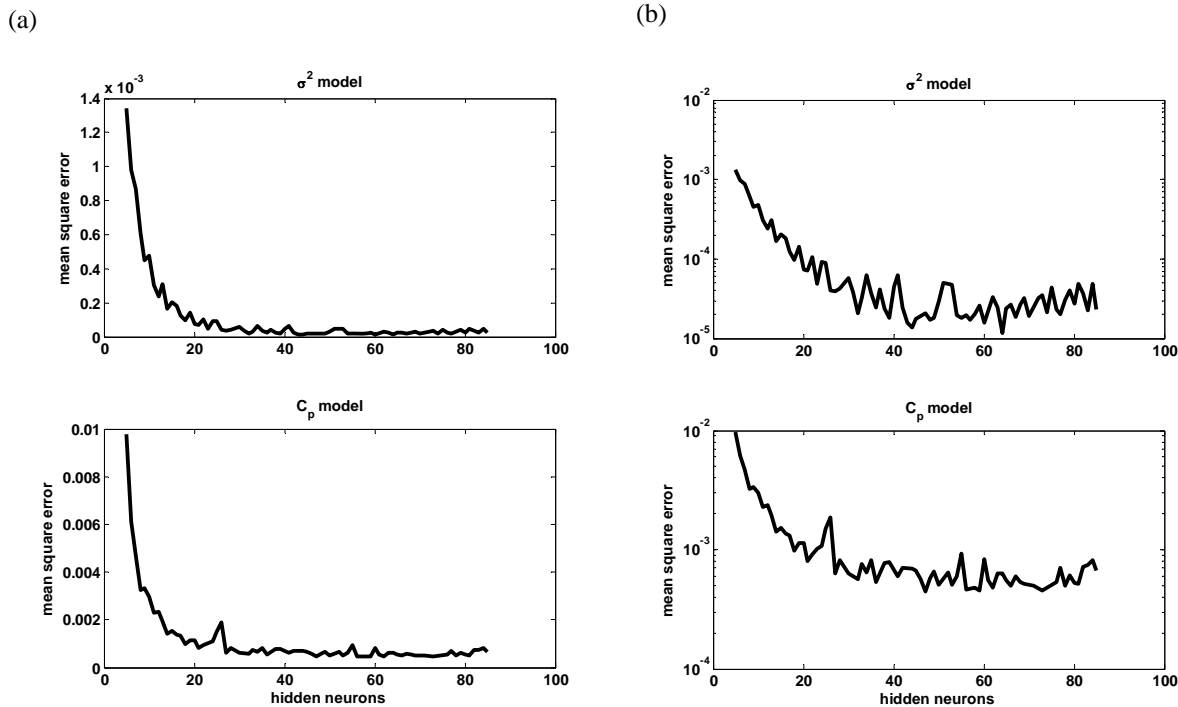


FIGURE 29. (a) Mean squared error results from cross KFCV model selection test. The minimum occurs near a network with a single layer of 50 neurons for both models. As the complexity of the network increases, the generalization error increases even though training error may decrease due to ‘overfitting’ the data. (b) The trend is easier to see with the data logarithmically scaled. Noise in the figures is from the randomness added by initializing the Levenberg-Marquardt training algorithm.

rons will generate a model with the best generalization error. Thus a pool of networks is created and trained at the optimal size, and the network with the lowest training error for our model is selected.

The performance of the cross injector network is measured by comparing to a validation test set. This test set includes simulations in the π_{3c} - π_{4c} plane for fixed values of $\pi_{1c} = 0.9$ and $\pi_{2c} = 0.9$. Fig. 30 shows the results of the neural network calculation compared to numerical solution of the governing PDE's to determine the injected band variance and peak height. The average relative error is 10.5%. Increasing the pool of optimally sized networks, or adding additional data points to the training set would be effective ways of reducing the error further.

Double-Tee Injector. The double-tee injector is similar to the cross injector, except for the offset between the loading channels. The most common value for this separation is two channel widths, however, for this example this value is a variable. The results of an injection at various steps in time, for two channel width separation, are shown in Fig. 31 The injector operates in two main stages, loading and dispensing. For the loading stage of this example, the analyte shaded in red enters the injector through port (A) and exits through port (C) into a waste reservoir. Simulta-

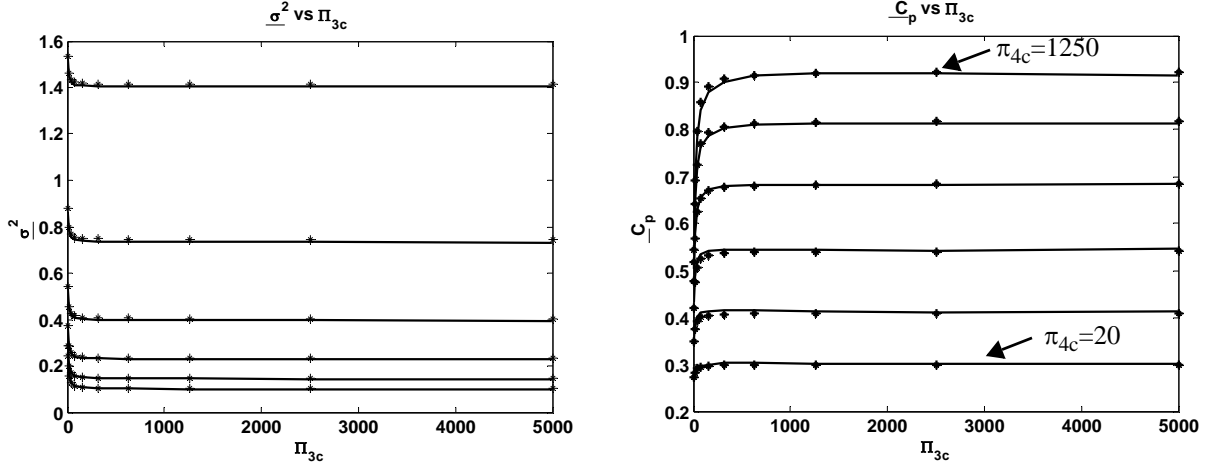


FIGURE 30. Plots of band variance and peak height vs. π_{3c} for varying values of π_{4c} . The maximum error of both models is approximately 10.5%. Dots are the numerical simulations, solid lines are the NN function results.

neously, accessory fields on the sides constrict the analyte on its sides from ports (B) and (D). After the analyte has completely filled the injection chamber, the applied voltages are changed for the dispensing stage so the electric fields push the analyte plug into the separation channel at port (B). While the main driving fields push the plug into the separation channel, accessory electric fields are created in the top and bottom channels at ports (A) and (C) to pull analyte back from the injection intersection, thereby preventing analyte leakage back into those channels while the plug is being injected.

The simulation parameters of the double-tee model are found in Table 4 and are nearly the same as the cross injector. The main difference is the double-tee has a parameter to describe the offset of the input channels π_{5d} . This parameter is varied from -2 to 2 to represent a backward to forward offset of two channel widths, the most common offset found in the double-tee injector. Fig. 32

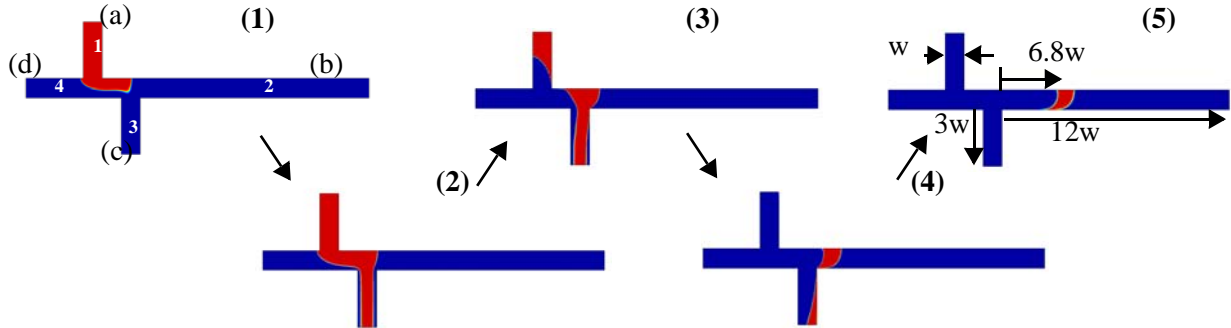


FIGURE 31. Snapshots of the transient injection process for the double-tee injector using electrokinetic transport. Starting on the left, the analyte in red is drawn from the sample reservoir at the top(1). After filling the injection chamber (2), the applied potential is changed to inject a plug of analyte while evacuating leftover sample to the north and south to prevent leakage (3,4,5).

Table 4. Double-Tee Simulation Parameters

| Simulation Parameters | System Non-Dimensional Parameters | Non-Dimensional Parameter Constraints |
|--|---|--|
| <u>Input</u> | <u>Input</u> | <u>Input</u> |
| μ - Electrokinetic mobility | $\pi_{1d} = E_{2L}/E_{1L}$ | $\pi_{3d}, \pi_{4d} \in [10, 5000]$ |
| κ - Diffusion coefficient | $\pi_{2d} = E_{1D}/E_{2D}$ | $\pi_{1d}, \pi_{2d} \in [0.01, 5]$ |
| $E_{1L}, E_{2L}, E_{3L}, E_{4L}$ - Loading electric field | $\pi_{3d} = \mu E_{3L} w / \kappa$ | $\pi_{5d} \in [-2, 2]$ |
| $E_{1D}, E_{2D}, E_{3D}, E_{4D}$ - Dispensing electric field | $\pi_{4d} = \mu E_{2D} w / \kappa$ | $E_{2L} = E_{4L}$ |
| w - Channel width | $\pi_{5d} = L_i / w$ | $E_{1D} = E_{3D}$ |
| C_o - Analyte concentration | | $\pi_{2d} > 1/16.65 \log(8490/\pi_{4d})$ |
| <u>Output</u> | <u>Output</u> | |
| C_p - Equivalent Gaussian Peak | $\underline{C}_p = C_p / C_o$ | |
| σ^2 - Equivalent Gaussian Variance | $\underline{\sigma}^2 = \sigma^2 / w^2$ | |

shows how the injector topology is modified with this parameter and its affect on the output plug, all other parameters held constant. As the parameter becomes equal to zero, the model describes the standard cross injector. This figure shows that a forward offset produces the largest band, followed by a zero offset and then the backward offset. This effect is due to the manner in which the injection chamber is loaded, and the magnitude of pullback for leakage control. Since the material is loaded from the top (1) and pinched from the sides (2,4), it has a higher concentration towards the top of the channel, as seen in step 2 of Fig. 31. As the plug travels into the separation channel (2), if it must pass the source channel (1), it will lose more mass than if it must pass the waste channel (3). This is due to the vacuuming effect of the pullback accessory fields in channels 1 and 3, and the fact that there is more mass on the top of the plug than on the bottom. Thus a backward offset causes the

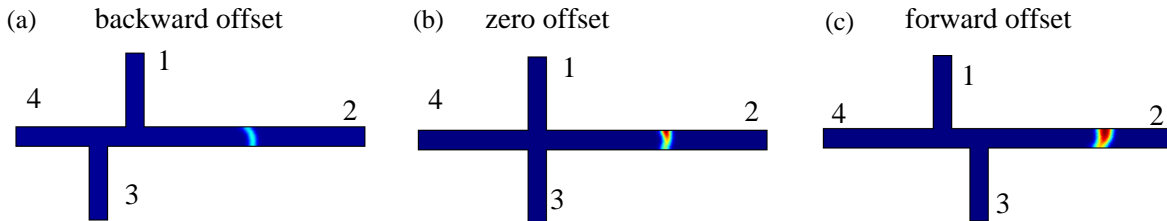


FIGURE 32. Simulations of the double-tee injector for $\pi_{1d} = \pi_{2d} = 2.5$, $\pi_{3d} = \pi_{4d} = 500$ for (a) $\pi_{5d} = -2$ (b) $\pi_{5d} = 0$ (c) $\pi_{5d} = 2$. The surface plots are not to scale, (a) and (b) are scaled to a maximum concentration of 0.5 and (c) is scaled to a maximum concentration of 1. (a) is a backward offset which produces the smallest plug, (b) is a zero offset, and (c) is a forward offset which produces the largest plug. This effect is apparent

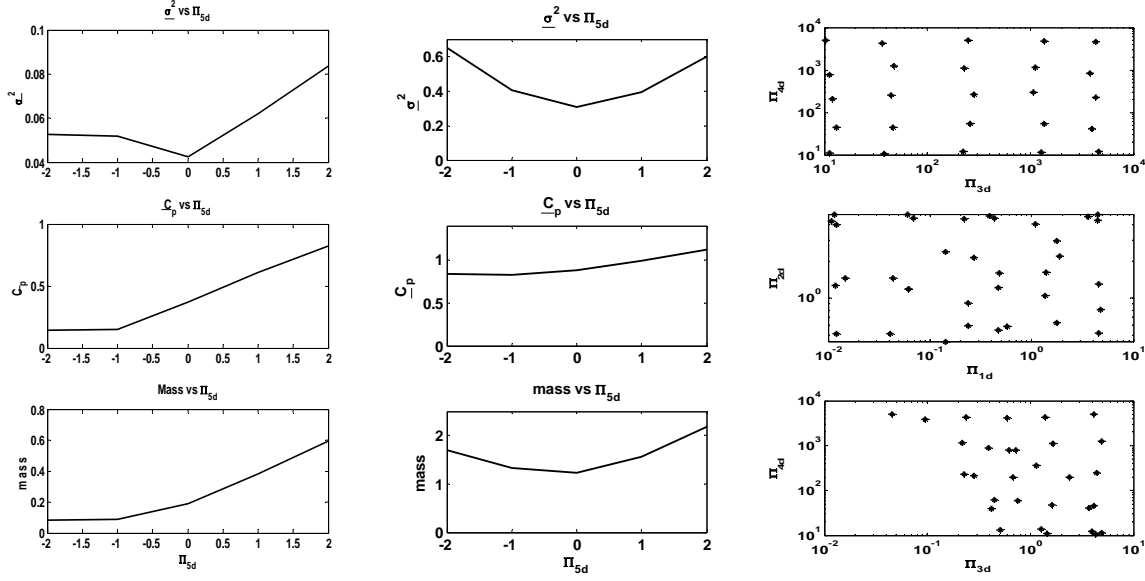


FIGURE 33. (a) Trace of the variance, peak height, and mass vs. channel offset for $\pi_{1d} = \pi_{2d} = 0.5$, $\pi_{3d}=400$, $\pi_{4d}=400$. (b) Trace of the variance, peak height, and mass vs. channel offset for $\pi_{1d} = \pi_{2d} = 2.5$, $\pi_{3d}=400$, $\pi_{4d}=400$. (c) Projected sampling of the design space.

band to lose more mass due to the pullback effect and a forward offset causes to lose the least amount of mass with a zero offset in the middle. This behavior is automatically accounted for in the numerical simulations used to train a neural network model, and is less significant for injections that use weak pullback fields.

As with the cross injector, traces in the design space can be constructed to get a sense of the structure of the response surface. The operation is similar to that of the cross, except for the offset parameter, π_{5d} . At nominal values of $\pi_{1d} = \pi_{2d} = 2.5$, $\pi_{3d} = 400$, $\pi_{4d} = 400$, the plot appears in Fig. 33a. When the effects of the pullback are not significant $\pi_{2d} < 1$, the result is more symmetric with the channel offset, as seen in Fig. 33b. Based on these plots π_{5d} is sampled on a linear scaling, and all other parameters are sampled on a logarithmic scale, as with the cross injector. The resulting sampled space is seen in Fig. 33c. A total of 4574 simulations were used to sample the space.

Because of the slightly more complicated response surface structure, a neural network with two hidden layers is used, as in Fig. 6b, [44]. The results of the KFCV model selection are seen in Fig. 34. For this test, the two layers were constrained to have the same size, leaving one-degree of freedom. The layers could be sized independently to create a two dimensional error surface, at the cost of additional computation. The increase in ‘overfitting’ error is again evident as the error

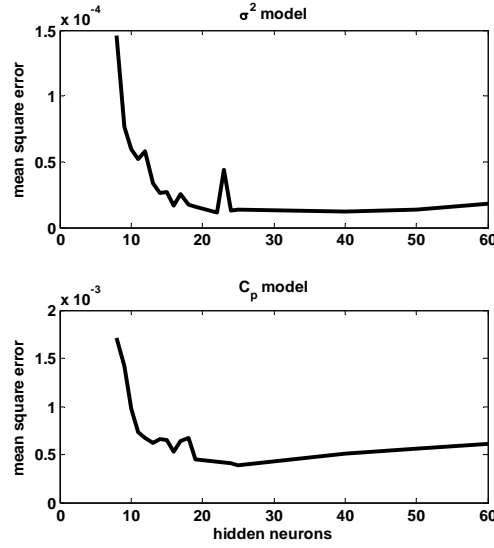


FIGURE 34. Mean squared error results from double-tee KFCV model selection test. The minimum occurs near a network with a single layer of 35 neurons for the variance model and 25 for the

decreases and then increases with increasing network size. The optimal network size for the variance model occurs near 35 nodes in each layer and near 25 nodes in each layer for the peak concentration model.

The performance of the double-tee injector network is measured by comparing to a validation test set. This test set includes simulations in the π_{3d} - π_{4d} plane for fixed values of $\pi_{1d} = 0.9$, $\pi_{2d} = 0.9$, and $\pi_{5d} = 2$. Fig. 35 shows the results of the neural network calculation compared to numerical solution of the governing PDE's to determine the injected band variance and peak height. The maximum relative error is approximately 7.72% for both the variance and peak concentration model. Increasing the pool of optimally sized networks, or adding additional data points to the training set would be the most effective ways of reducing the error further.

Gated Cross Injector. The gated-cross, Fig. 36, has the same geometry as the cross, but uses a three stage operating scheme. The gated-cross control scheme allows for continuous flow injection, so that new sample can be loaded at the same time that previously dispensed plugs are run through the separation channel. The first step of operation involves creating the gate in the injection chamber by counterflowing an analyte stream against a buffer stream. The second step involves removing the applied potential from the buffer port, which allows a portion of analyte to overflow the injection chamber. The final step is a return to the applied potentials of the first stage, which reestablishes the gate, while simultaneously injecting the overflowed analyte into the separation channel.

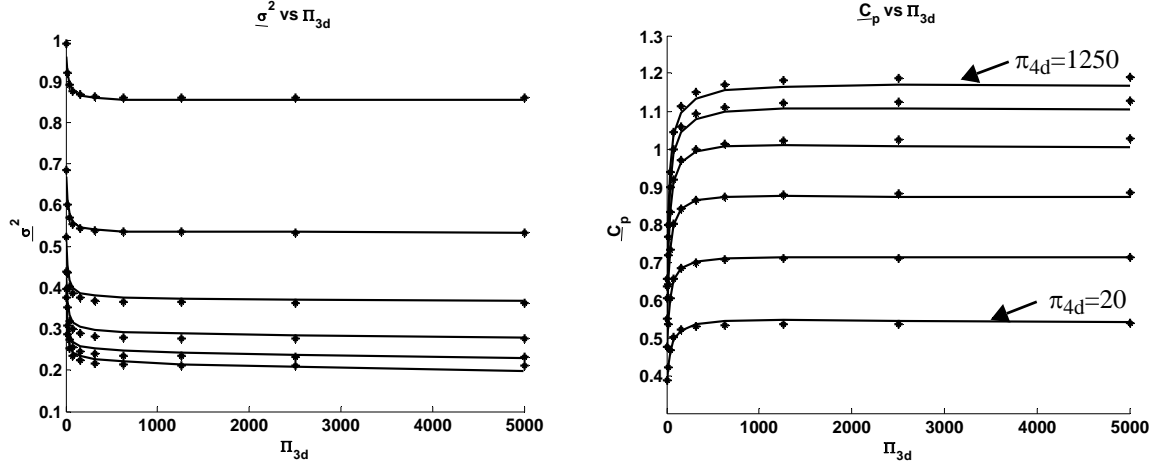


FIGURE 35. Plots of band variance and peak height vs. π_{3c} for varying values of π_{4c} . The maximum error for both models is approximately 7.72%. Dots are the numerical simulations, solid lines are the NN function results.

Since the action taken in the second stage involves floating only one node while all others remain unchanged, no independent parameters are introduced by this stage. The fields and flow patterns are completely determined by the parameters set in the first stage.

As a result of this different operation the gated-cross has a set of parameters that differ significantly from those of the cross and double-tee, as seen in Table 5. The first parameter, π_{1G} represents the extent to which the gate is closed. As discussed in [76], as long as $E_1 \geq E_2$, the gate will remain closed in the limit of no diffusion. As π_{1G} is reduced, the gate is further closed, so in practice $E_1 > E_2$ in the presence of diffusion. The second parameter represents the ratio of the buffer

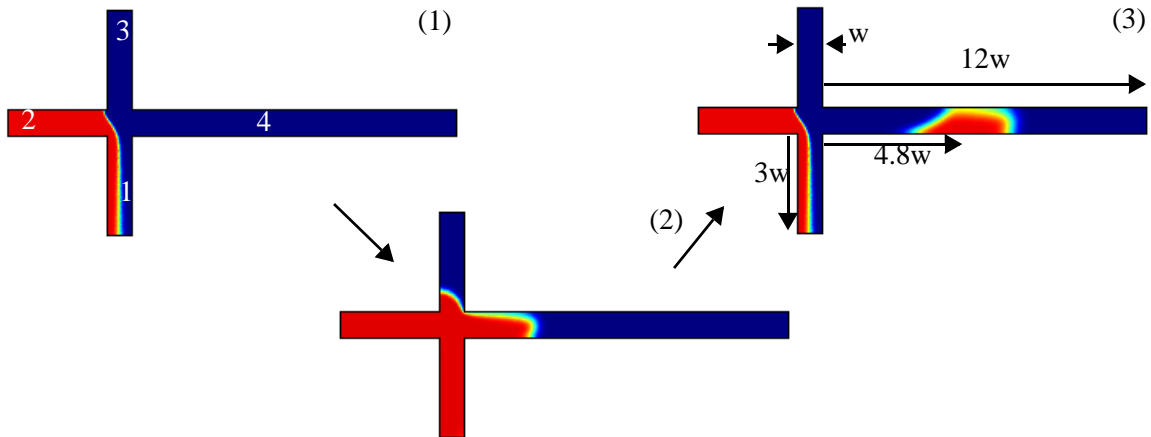


FIGURE 36. Snapshots of the transient injection process for the gated-cross injector using electrokinetic transport. Starting on the left, the analyte in red is drawn from the sample reservoir at the top and takes a 90° turn forming a gate in the injection chamber(1). The chip operates in this mode until a plug is needed. At this time the electric potential in channel 3 is floated causing the analyte to flood the chamber (2). When the plug of desired size is formed, the potential is reapplied and the gated is reformed injecting the plug into the separation channel (3).

electric fields (E_3, E_4) to the analyte electric fields (E_2, E_1). As π_{2G} increases, the buffer fields become larger relative to the analyte fields. The third parameter, π_{3G} , represents the Peclet number of the loading phase. The final dynamic parameter, π_{4G} , is a measure of the ratio of length of the injected plug, as related to the floating time, T_{LD} , to the channel width.

As with the cross and double-tee, a significant concern in the operation of a gated-cross injector is the leakage that can occur in the separation channel if the gate is not sufficiently closed, as seen in Fig. 38. If the leakage is too great, the injector will not operate because the increased noise floor will make a separation impossible. A region of feasibility must be determined within which the injector can be modeled. This leakage was analyzed in [76] as a function of the gate closure and the system Peclet number. They determined a boundary indicated by 1% leakage of the flux of the analyte from the source reservoir into the separation channel. In this work, we determine the boundary for a more complete set of physical parameters to create a region of feasibility.

This region of feasibility is determined with a classification neural network. For classification networks, it has been shown that a decision boundary of arbitrary complexity can be defined using only three layers (two hidden, one output) [44]. Using the simulations like those in Fig. 38, a network with a topology of Fig. 6b with two hidden layers each with two neurons, is created to define the region of feasible operation. Fig. 37 shows the feasibility regions for the gated-cross. The

Table 5. Gated Cross Simulation Parameters

| Simulation Parameters | System Non-Dimensional Parameters | Non-Dimensional Parameter Constraints |
|---|---|---------------------------------------|
| <u>Input</u> | <u>Input</u> | <u>Input</u> |
| μ - Electrokinetic mobility | $\pi_{1g} = E_2/E_1$ | $\pi_{1g} \in [1/10, 1]$ |
| κ -Diffusion coefficient | $\pi_{2g} = E_3/E_1$ | $\pi_{2g} \in [1, 4]$ |
| $E_{1L}, E_{2L}, E_{3L}, E_{4L}$ - Loading electric field | $\pi_{3g} = \mu E_3 w / \kappa$ | $\pi_{3g} \in [10, 5000]$ |
| w - Channel width | $\pi_{4g} = w / \mu E_{2D} T_{LD}$ | $\pi_{4g} \in [1/20, 1/3]$ |
| C_0 - Analyte concentration | | |
| <u>Output</u> | <u>Output</u> | |
| C_p - Equivalent Gaussian Peak | $\underline{C}_p = C_p / C_0$ | |
| σ^2 - Equivalent Gaussian Variance | $\underline{\sigma}^2 = \sigma^2 / w^2$ | |

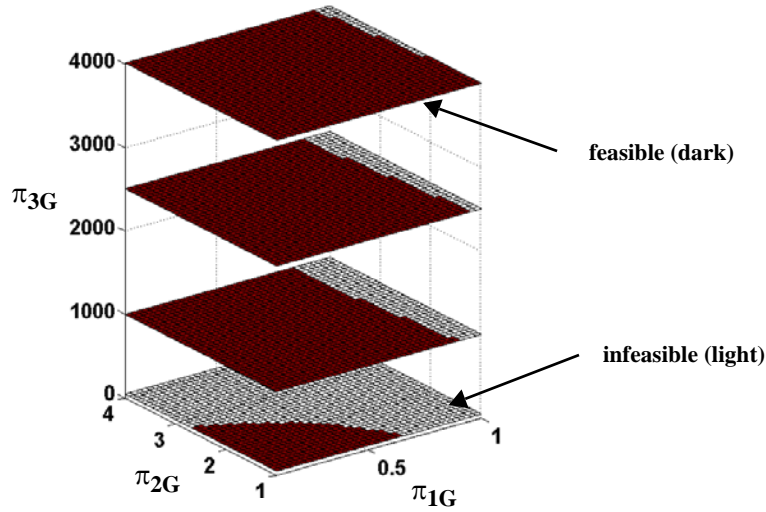


FIGURE 37. Gated-cross feasibility space (dark is feasible)

feasibility space is independent of π_{4G} because the leakage is determined in the loading stage independently of how the band is dispensed. The most notable feature of the feasibility space is that as π_{3G} , the Peclet number, becomes smaller, the amount of feasible space decreases significantly. This is due to the fact that for a given gate-closure, when diffusion is large the diffusive flux leaking into the separation channel is large.

The gated-cross model is sampled in a similar fashion as the cross and double-tee. The Peclet number is sampled on a logarithmic scale, all others are linear. The points in the infeasible region are moved into the feasible region using the same Gaussian resampling algorithm with the cross and double-tee. For a network with a single hidden layer of neurons, the resulting optimal model is chosen using the KFCV algorithm, Fig. 39a The optimal model sizes are approximately 35 hidden nodes for both the variance and peak concentration models. As a qualitative measure of the

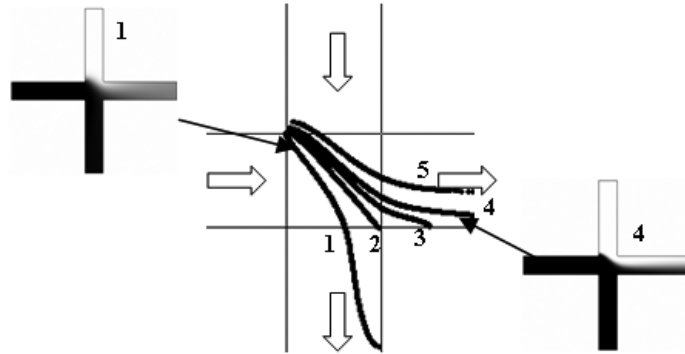


FIGURE 38. Gated-cross leakage tested at Peclet numbers from 19 at contour 1 to ∞ at contour 5. The contours are defined by the 7% of the maximum concentration, and agree very well with the numerical and experimental results found in [76]

accuracy for the gated-cross model, the response surface results in the π_{1G} - π_{2G} plane are shown in Fig. 39b and Fig. 39c for the variance and peak concentration models respectively. The NN model can generate a much denser surface of points in a fraction of second compared to the hours it takes a numerical simulation to generate a much less dense surface. This type of computational efficiency is what enables simulation and synthesis for design, where many repeated simulations may be required.

4.1.6. Micro Reactor Model

Traditional chemical manufacturing is heavily based on economy of scale with large reactors and associated plants requiring large process batches and associated large scale transport and storage of raw materials and products. All these large scale features present health and safety problems which can lead to major disasters as well as unacceptable levels of risk to operators and the neighboring community [85]. Microreactor chemistry shows great promise as a novel method on which to build new chemical technology and processes. Micro reaction systems are fabricated using techniques originally developed for electronic circuits. In their simplest form, micro reactor devices consist of network of micron-sized channels, typical dimensions are in the range of 30-300 μm ,

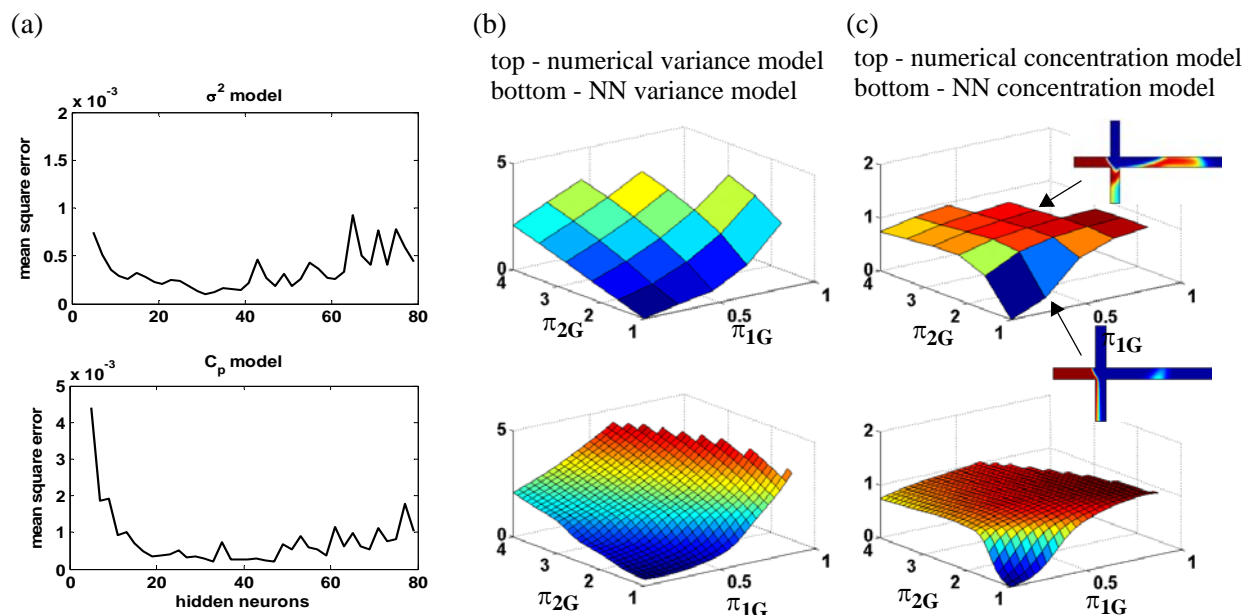


FIGURE 39. (a) Mean squared error results from gated-cross KFCV model selection test. The minimum occurs near a network with a single layer of 35 neurons for the variance and concentration models. (b) Response surface of numerical simulation on top and neural network on bottom for variance model. The neural network is able to produce results in a fraction of second, whereas the numerical simulation takes hours. (c) Response surfaces for peak concentration model, numerical simulation on top, neural network on bottom. π_{3G} and π_{4G} are fixed at 232 and 0.1208 respectively.

etched into a solid substrate [85],[86],[87],[88]. For solution-based chemistry, the channel networks are connected to a series of reservoirs containing chemical reagents and products. Reagents can be brought together in a specific sequence, mixed and allowed to react for a specified time in a controlled region of the channel network using electrokinetic or hydrodynamic pumping. We consider electrokinetically driven systems in our work. In these systems electrodes are placed in the appropriate reservoirs to which specific voltage sequences can be delivered under automated computer control. This control offers a simple but effective method of moving and separating reactants and products within a microreactor, without the need for moving parts. It enables the ability to manipulate reagent concentrations in both space and time within the channel network and provides an additional level of reaction control which is not attainable in bulk stirred reactors where concentrations are generally uniform. Furthermore, the spatial and temporal control of chemical reactions in micro reactors, coupled with the features of very small reaction volumes and high surface interactions, is akin to the situation of reactions within biological cells.

Much of the current research on microfluidic reaction systems is centered on the fabrication of micro reactors for experimentation. Ehrfeld *et. al.* [86] presented an overview of the manufacturing techniques that covered the fabrication of micro mixers, micro reactors, and micro heat exchangers. Jensen *et. al.* [89] created a hydrodynamically pumped cross-flow micro reactor and demonstrated the potential of it as a laboratory tool for heterogeneous catalyst testing. Haswell *et. al.* [90] considered electrokinetic based fluidic pumping systems. They described the fundamental characteristics and emerging applications of micro reactors in the field of synthetic chemistry [85][91]. While the micro reaction systems are undergoing dramatic expansions, the computer-aided design (CAD) tools have not kept pace.

In this project we developed two reduced order models for electrokinetic continuous-flow micro reactors. The models can take any concentration profile of reactants at the reactor inlets and simulate diffusion, convection, and reaction inside of reaction channels of any length. Both approaches reduce the set of Partial Differential Equations (PDEs) to a system of coupled Ordinary Differential Equations (ODEs). Our first approach is based on physical insights and consists of approximating the reactive channels by a series of interconnected parallel Plug Flow Reactors (PFR). The second approach is based on mathematical simplification of the underlying equations. By using the Method of Lines (MOL), the spatial derivatives for the diffusional mixing in reaction channels are approximated by finite difference relationships. These models are verified by numeri-

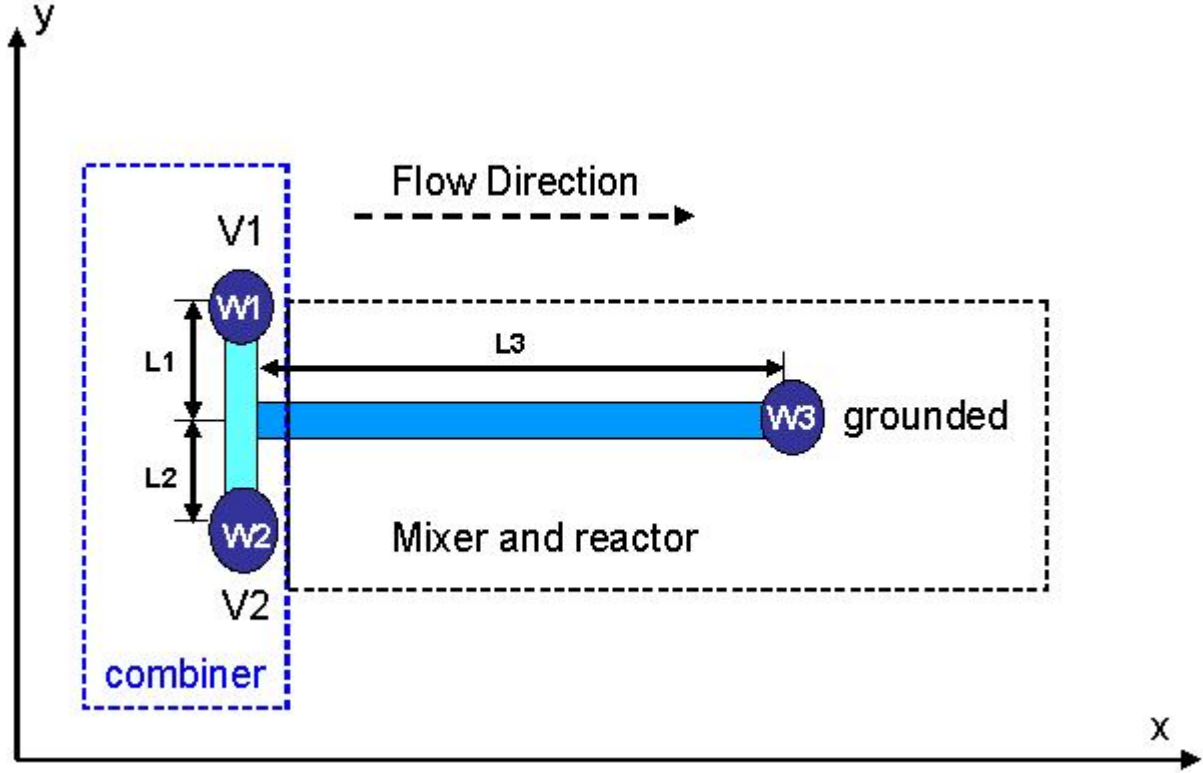


FIGURE 40. Schematic illustration of a basic micro mixing and reaction system.

cal simulations in COMSOL, a finite element solver [92]. These models are orders of magnitude faster than simulations in COMSOL, and therefore allow parametric studies of a particular micro reactor as well as enable integration into total LoC synthesis and layout approaches [93].

In the reaction channel as shown in Fig. 40, the fluid driven by electric field only flows in the x direction and the species' velocity profiles across the channel are flat except close to the channel wall [94][95]. As a result, \mathbf{v}_i only has one component v_{ix} and convection happens in the x direction only. Since the species are not always perfectly mixed before entering the reaction channel, their concentrations may vary in both the x direction and the orthogonal y direction across the channel. The low Reynold number operation of micro scale flows [96] indicates that mixing in the y direction is caused by diffusion only. In a typically designed microfluidic system, the fluid flows fast enough that the concentration changes caused by diffusion in the x direction is much less than those caused by convection. In channel based micro reactors, it is safe to only consider the diffusion in the y direction and ignore the diffusion in the x direction. As we briefly mentioned earlier, there are two operation stages for LoC system and mixing and reaction are belong to the steady-state

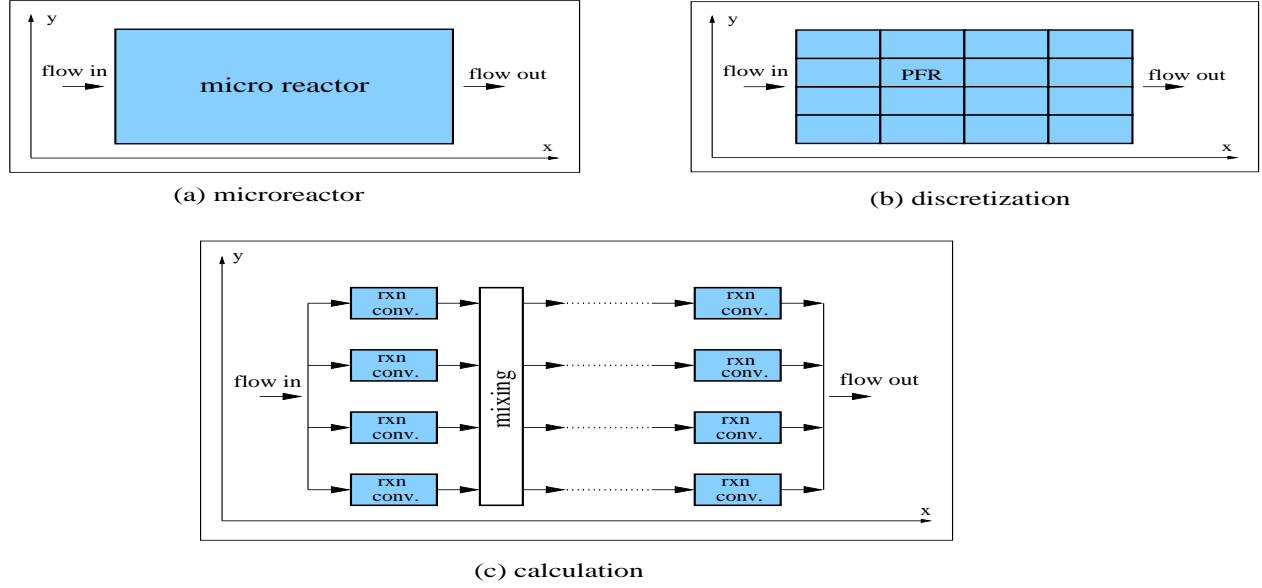


FIGURE 41. The micro reactor is represented by a PFR network model. The calculation of reaction and convection is conducted in one column of PFRs and the calculation of diffusional mixing is conducted between two columns of PFRs.

loading stage. We further assume that all diffusion coefficients are independent of concentration and that all thermal effects are negligible. After applying these assumptions, we have:

$$v_{i,x} \frac{\partial c_i}{\partial x} = D_i \left(\frac{\partial^2 c_i}{\partial y^2} \right) + r_i \quad (38)$$

Our micro reactor models are based on this set of PDEs. Here we present two different approaches that approximate this PDE system as an ODE system which is easier to solve and normally has shorter computational time.

Plug Flow Reactor Model. In our microfluidic system, the fluid has a plug-like velocity profile. When modeling this system, it is natural to base the model on the classic Plug Flow Reactor (PFR). To capture the real behavior of the reactive fluid in micro reactors, we discretize our system and use multiple PFRs to model the micro reactor. The discretization and calculation of the PFR network model is shown in Fig. 41. This network has M rows and N columns. Each PFR in the network has the length of l and the width of w . We use $P_{m,n}$ to stand for the small PFR in the m th row and n th column. The concentrations of the species inside $P_{m,n}$ can be represented as $c_{m,n}^i(x)$, where the range of x is from 0 to l . The total volume of the PFRs in the network remains the same as the orig-

inal micro reactor by enforcing $\sum_m w = W$ and $\sum_n l = L$. In this PFR network model, the calculation of concentration change is conducted alternatively in the x and y direction.

- **Step 0:** When $n = 1$, calculate the initial concentrations for the first column of PFRs $c_{m,1}^i(0)$ by assuming perfect pre-mixing for each PFR and average the inlet concentrations associated with $P_{m,1}$.
- **Step 1:** Eq. (38) with the diffusion term zeroed out is used to calculate convection and reaction inside each PFR $P_{m,n}$ to generate $c_{m,n}^i(x)$. When $n=1$, the initial concentrations $c_{m,n}^i(0)$ are calculated by Step 0; otherwise, they are set to the values of the modified outlet concentrations from the previous column of PFRs $P_{m,n-1}$:

$$c_{m,n}^i(0) = c_{m,n-1}^i(l) \quad (39)$$

- **Step 2:** Calculate the average concentration $\overline{c_{m,n}^i}$ inside $P_{m,n}$:

$$\overline{c_{m,n}^i} = \frac{\int_0^l c_{m,n}^i(x) dx}{l} \quad (40)$$

This average concentration is used in the next step for approximating the diffusional mixing.

- **Step 3:** Fick's second diffusion law is used in the y direction between two columns of PFRs to approximate the concentration change caused by the diffusional mixing. We approximate the second derivative with finite difference:

$$\Delta c_{m,n}^i = D \cdot \frac{\overline{c_{m-1,n}^i} - 2\overline{c_{m,n}^i} + \overline{c_{m+1,n}^i}}{w^2} \cdot \frac{1}{v_i} \quad (41)$$

- **Step 4:** Calculate the modified outlet concentration for $P_{m,n}$:

$$c_{m,n}^i(l) = c_{m,n}^i(0) + \Delta c_{m,n}^i \quad (42)$$

Then the calculation will repeat from *Step 1* to *Step 4* until reaching the final column. The combination of the two parameters M and N that produces accurate results for a system with given Peclet number (Pe) and Damkohler number (Da) is not unique. A fine mesh can lead to a result with certain accuracy, but requires long computational time.

Method of Lines Model. Eq. (38) is the governing equation for our micro mixing-reaction system. We need to solve the set of PDEs. In this section, we develop the micro reactor model using the numerical method of lines (MOL), which is a technique for solving partial differential equations by discretization all but one dimension, and then integrating the semi-discrete problem as a system of ODEs [97]. A significant advantage of the method is that it allows the solution to take advantage of

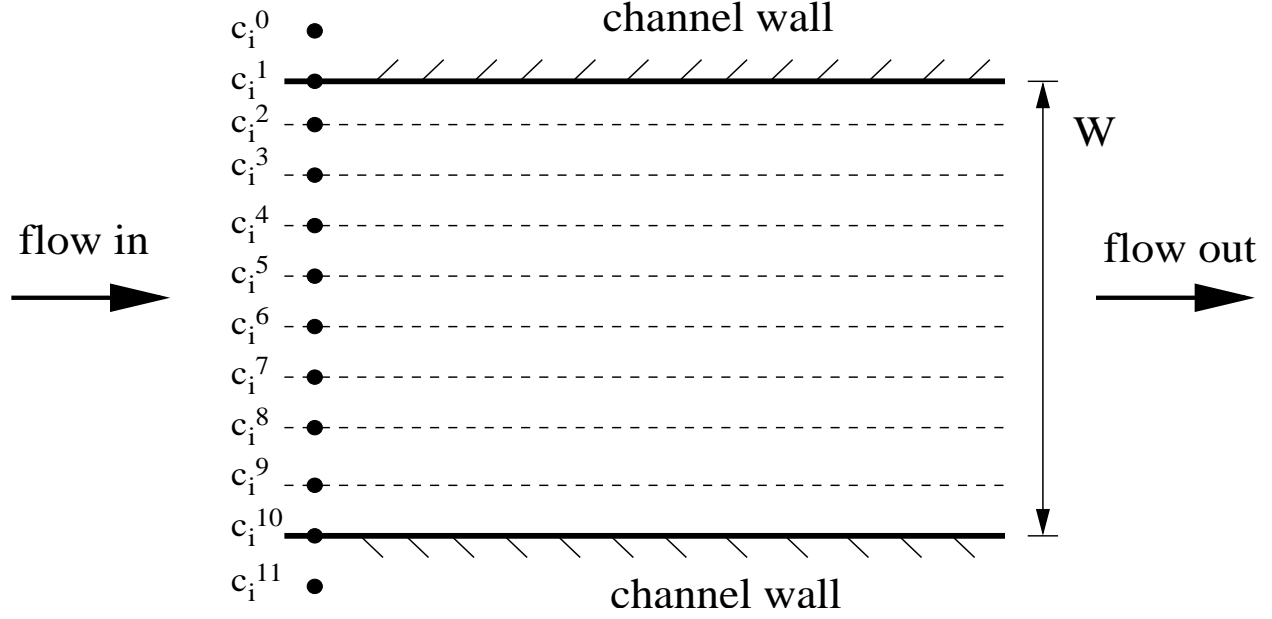


FIGURE 42. Apply the numerical method of lines to calculate the diffusion in the channel width direction. Here, we take 10 discretized concentration points in the channel: $c_i^1 - c_i^{10}$. Points for c_i^0 and c_i^{11} are needed to apply the boundary conditions.

the sophisticated general purpose methods and software that have been developed for numerically integrating ODEs. It is necessary that the PDE problem be well-posed as an initial value problem in at least one dimension. In our micro systems, our set of PDEs satisfy this requirement.

We discretize the system and take M concentration points along the channel width as shown in Fig. 42, in which $M=10$. The second order centered formula based on the Taylor expansion can be used to approximate the first and the second derivative:

$$f(x^k) = \frac{f(x^{k+1}) - f(x^{k-1})}{2h} + O(h^2) \quad (43)$$

$$f'(x^k) = \frac{f(x^{k+1}) - 2f(x^k) + f(x^{k-1}))}{h^2} + O(h^2) \quad (44)$$

where k indicates the number of discretization elements. The initial conditions for the set of ODE are decided by assuming perfect pre-mixing and average the inlet concentrations for each discretization. As the micro fluid system is driven electrokinetically, the boundary conditions for the fluid flow are treated as slip boundary conditions, which means the species has the same velocity v_{ix} at each point. The Neumann boundary condition [98] is used for the concentration change at the channel walls:

Table 6. Timing results for the PFR network model.

| MXN | 2x2 | 4x3 | 4x4 | 6x5 | 6x9 | 8x9 | 10x9 | 10x10 |
|--------------|-------|-------|-------|-------|-------|-------|-------|-------|
| error (%) | 16.7 | 8.8 | 3.8 | 2.6 | 1.9 | 0.7 | 30.8 | 0.2 |
| cpu time (s) | 0.058 | 0.060 | 0.072 | 0.081 | 0.107 | 0.121 | 0.143 | 0.156 |

$$\frac{\partial c_i^1}{\partial y} = 0 \text{ and } \frac{\partial c_i^M}{\partial y} = 0 \quad (45)$$

By using Eq. (43), these two boundary conditions can be expressed as $c_i^0 = c_i^2$ and $c_i^{M+1} = c_i^{M-1}$. Then the original set of PDEs (Eq. (38)) can be reformulated into a set of ODEs as follows:

$$v_{ix} \frac{\partial c_i^m}{\partial x} = D_i \left(\frac{c_i^{m-1} - 2c_i^m + c_i^{m+1}}{\Delta y^2} \right) + r_i \quad m = 2, 3, \dots, M-1 \quad (46)$$

$$v_{ix} \frac{\partial c_i^1}{\partial x} = D_i \left(\frac{2c_i^2 - 2c_i^1}{\Delta y^2} \right) + r_i \quad (47)$$

$$v_{ix} \frac{\partial c_i^M}{\partial x} = D_i \left(\frac{2c_i^{M-1} - 2c_i^M}{\Delta y^2} \right) + r_i \quad (48)$$

where Δy is the discretized channel width, which satisfies $\sum_m \Delta y = W$.

Increasing the amount of discretization results in higher accuracy but longer computation time. This set of ODEs can readily be solved by a standard ODE solver. We use the solver called Lsode to solve the set of ODEs.

Reaction Model Verification. To verify the accuracy of our models, we developed a partial differential equation based model using finite element methods in COMSOL and compared the results with our discretized PFR network model and MOL model. The case study involves a competitive reaction system with species A and B competing to react with a tag T to produce products AT and BT. The concentration profiles of the reaction product BT generated at the reactor outlet by FEM simulation is shown in Fig. 43(a), and the timing results as a function of element size are shown in Fig. 43(b).

The results from the PFR model are shown in Table 6. It takes less than 0.2 seconds to generate a result with error less than 1%, compared to the few minutes required for similar accuracy

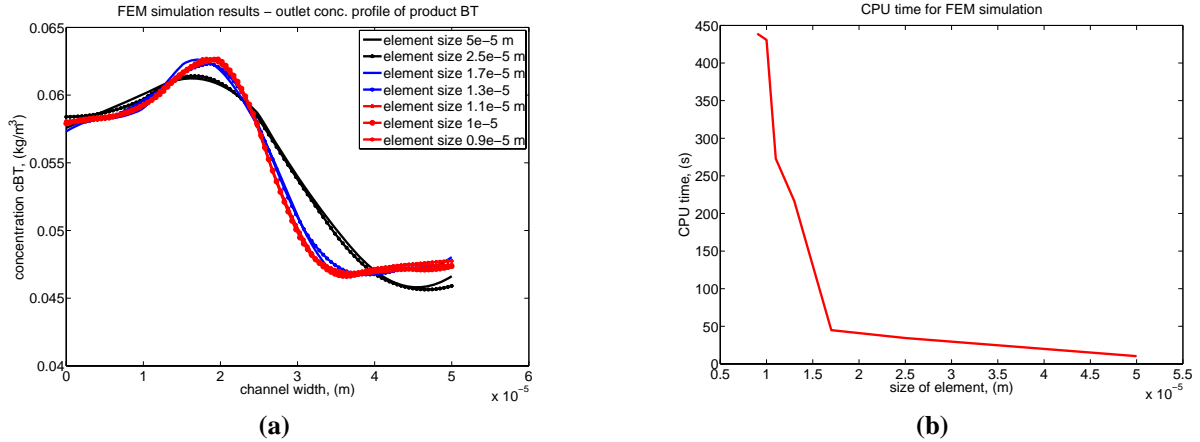


FIGURE 43. (a) The concentration profiles generated by FEM simulations with decreasing size of elements. The red curves represent the results associated with the small element size. They are grid independent and can be used as a basis for comparison. (b) Timing results for the FEM simulations.

from the finite element simulation.

The results for the method of lines (MOL) simulation for the AT product concentration and the timing results are shown in Fig. 44. The simulation result with 40 discretizations takes less than 0.25 seconds to generate an accurate result, i.e. three orders of magnitude faster than the FEM simulation (273 seconds).

4.2. System Simulation

This section demonstrates the integration of the behavioral models for subsystems (e.g., mixers in Section 4.1.1 and separation microchips in Section 4.1.2.) into a system-level schematic of integrated LoCs. Such a schematic can be iteratively simulated to capture the overall effects of

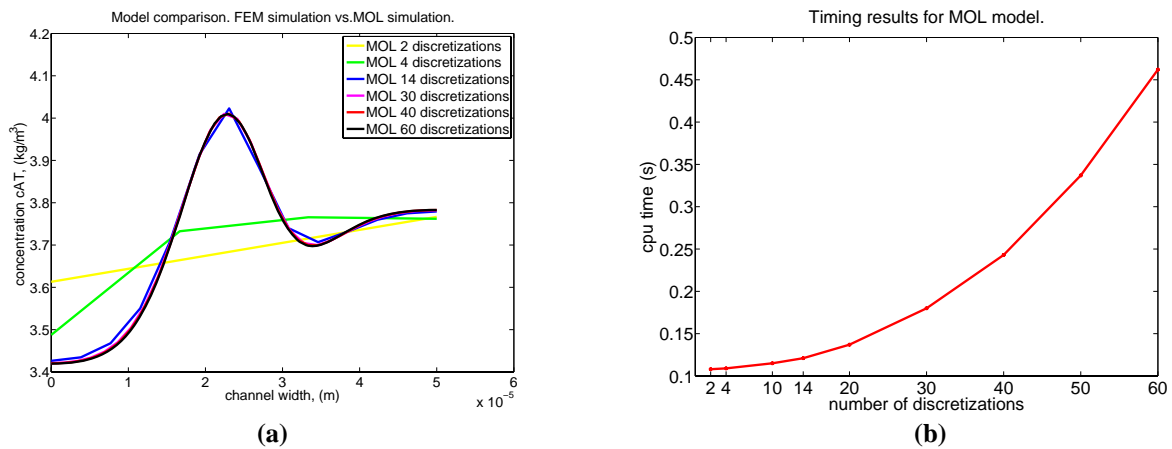


FIGURE 44. (a) Comparison of MOL model simulation results with FEM simulations. Using more discretization in the MOL model leads to a more accurate solution. (b) Timing results for the MOL model simulation as a function of the discretization.

subsystem-level parameters (e.g., separator and mixer topologies and types; operational electric fields and injection schemes etc.) and element-level parameters (e.g., length and width of mixing and separation channels etc.) on system performance, as well as the trade-offs among them, leading to a system-level optimal design achieved by “negotiation” among each subsystem and element.

Integrated LoC design is difficult because of the need to,

- Accurately interpret the fundamental multi-physics phenomena (such as electric, fluidic, heat transfer and sample transport) at element or component levels.
- Properly classify, decouple and reduce the multi-physics phenomena to tractable mathematical models, while still maintaining their essential physical characteristics.
- Efficiently integrate low-level mathematical models to obtain a system-level representation for iterative simulation-based design.

Most of the previous approaches to LoC modeling are not amenable to accurate simulation-based iterative design of integrated systems due to various limitations, such as difficulties in system-level representation, inaccurate physical descriptions and inflexibilities with respect to geometrical perturbations. Currently, the only general and accurate approach is the classical method of numerical simulation, which however suffers from unacceptable computational cost. Therefore, a methodology that simultaneously takes into account of balancing accuracy and efficiency without sacrificing generality is needed.

To demonstrate the ability of the methodology developed in this project to meet these goals, this section describes how it can be used to design a competitive immunoassay microchip consisting of mixing, reaction, injection and separation subsystems.

4.2.1. Integrated Competitive Immunoassay Microchip

Fig. 45a illustrates an integrated competitive immunoassay microchip proposed by Harrison *et al.* [22], which consists of four subsystems (mixing, reaction, injection and separation). Its operation involves both synthesis and analysis, which are typical functions performed in a biochemical lab. In the first phase, a negative electrical voltage is applied at reservoir 3 with reservoir 5, 6 and 7 grounded and the other reservoirs left floating. Theophylline (Th, from reservoir 5) and fluorescein-labeled theophylline tracer (Th^* , from 6), driven by the electric field, first mix with each other within channel J1-J3. Then Th and Th^* in the mixture compete for a limited number of antibody (Ab, from reservoir 7) binding sites in the mixing and reaction channel J3-J4. The mixture (hereafter called analyte) of reaction products Ab-Th^* and Ab-Th , as well as unreacted Th^* and Th are

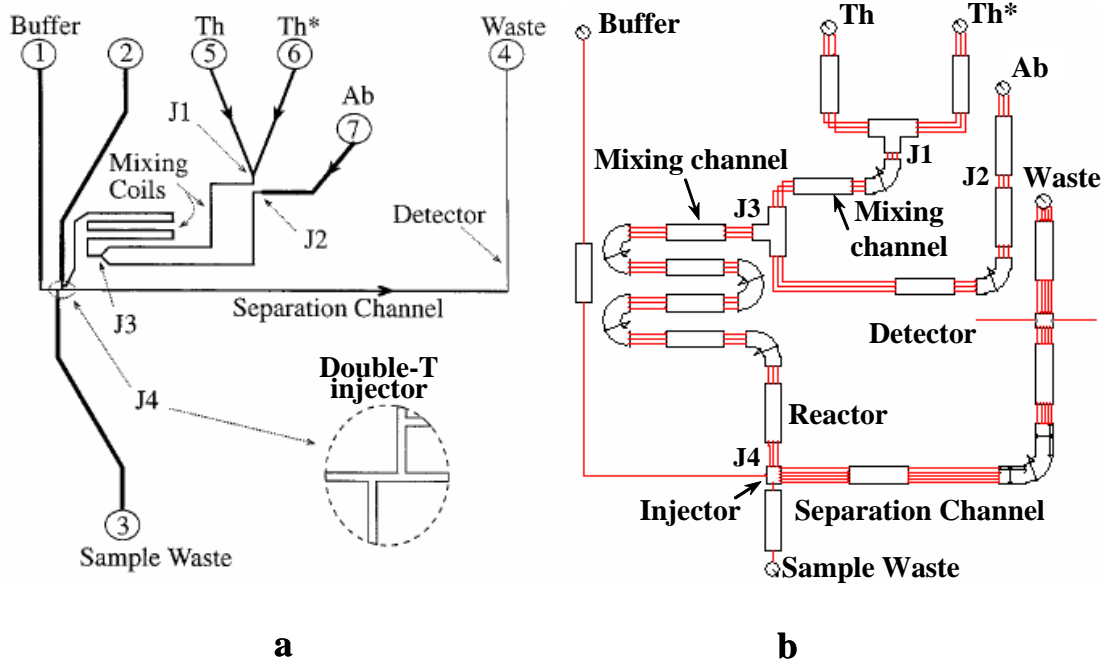


FIGURE 45. (a) Sketch of a competitive immunoassay microchip consisting of four subsystems (mixing, reaction, injection and separation) [22]. (b) Its system-level schematic.

loaded into the double-T injector. Th, Th* and Ab are continuously supplied by the reservoirs; therefore concentrations of all the samples, reagents and products in the mixer and reactor at this phase are in steady-state. This completes the synthesis operation.

In the second phase (analysis), the voltages are switched on reservoirs 1 and 4 with the others left floating. Thus the loaded analyte is injected as a band into the separation channel J4-reservoir 4. Because the analyte includes tagged Ab-Th* and Th* molecules that have different charges and sizes, they move at different speeds and eventually can be separated by electrophoresis [62]. The amount of Ab-Th* and Th*, represented by the areas under the electropherogram, for a wide range of Th concentration, can be obtained to generate a calibration curve to determine unknown concentrations of Th in the sample at reservoir 5 for clinical analysis. As the Ab-Th complex and Th are not tagged with fluorescent tracers, they are invisible to detection and not considered in the remaining operation. In this phase, Ab-Th* and Th* bands broaden due to molecular diffusion and other dispersion sources; therefore the transient evolution of their band shapes, directly impacting the analysis quality is of prime importance.

To simulate such a LoC, a system-level schematic that assembles behavioral element models based on the LoC geometric and functional hierarchy needs to be built. Fig. 45b illustrates the hier-

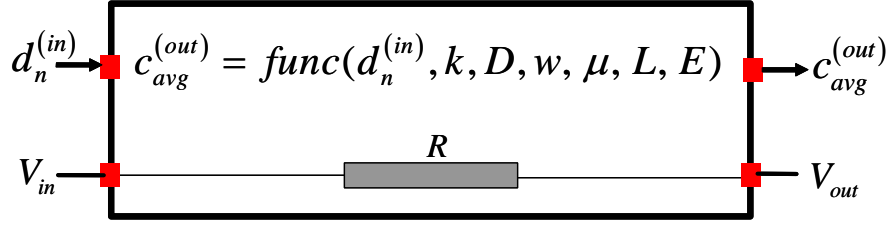


FIGURE 46. Behavioral model structure for the electrokinetic reactor. At the inlet, the Fourier coefficients ($d_n^{(in)}$) of widthwise concentration profiles of samples and reagents conveyed from upstream mixers characterize their premixing degree. At the outlet, biofluidic pins quantify the average concentrations ($c_{avg}^{(out)}$) of the reaction products and unreacted samples.

archical decomposition of the LoC. It is decomposed into four functional subsystems and further broken down into commonly used elements, such as the straight mixing channel, injector, reactor and turn separation channel.

4.2.2. Behavioral Models for Reactors and Injectors

In this section, behavioral models of competitive immunoassay reactors and connection pins (I/O parameters) of double-T injectors that are integral to the schematic in Fig. 45b will be presented.

Competitive Immunoassay Reactors. The biofluidic pins for a general purpose reactor will be defined to enable the extension of our system-level simulation approach beyond a competitive immunoassay design. The behavioral model for the reactor that will be developed is specific to the competitive immunoassay.

Pin Definition. In practical integrated LoCs, the reactor fulfills two functions, bridging the mixer and injector, as well as converting samples and reagents into reaction products (synthesis). Therefore, the biofluidic pins at its input and output terminals are different. Fig. 46 shows the behavioral model structure for the electrokinetic reactor. In addition to electric pins (V_{in} and V_{out}), biofluidic pins are also proposed with arrows indicating the direction of signal flow for calculating pin values. At the reactor inlet, the Fourier coefficients ($d_m^{(in)}$) of the widthwise concentration profiles of the samples and reagents input from upstream mixers characterize their premixing degree. At the outlet, biofluidic pins are defined as the average concentrations ($c_{avg}^{(out)}$) of the reaction products and unreacted samples that can be used by the downstream injector to determine the band shape of the injected species. Here, indices *in* and *out* represent the quantities at the inlet and outlet respectively.

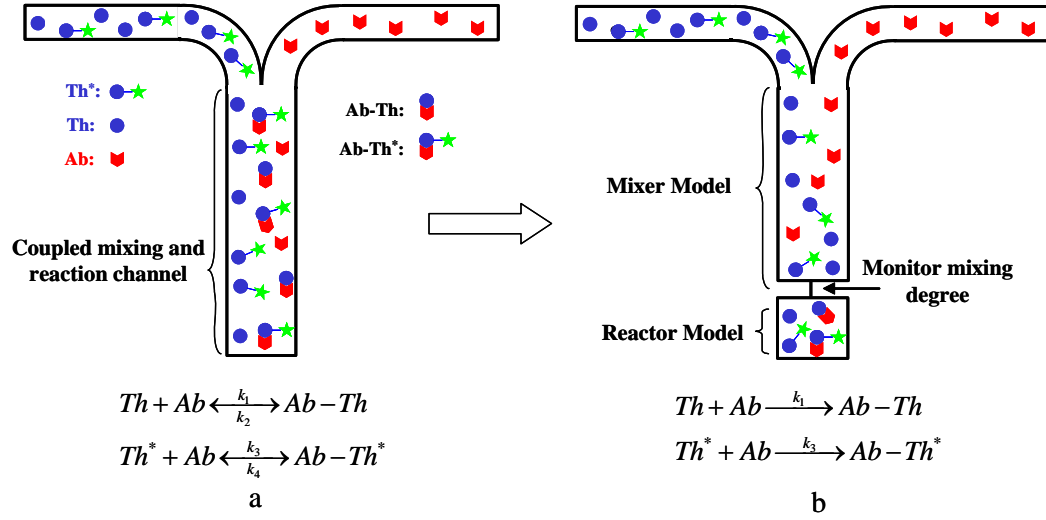


FIGURE 47. (a) Sketch of the competitive immunoassay reaction between Th , Th^* and Ab . (b) Sketch of the competitive immunoassay reaction model.

Analog wiring buses carrying vector-type pin values (concentration coefficients or multiple samples and reagents) similar to those used in mixers and separators are also employed.

Behavioral Model. Fig. 47 depicts the process of competitive immunoassay reactions between Th , Th^* and Ab , as well as the model we use. The uniformly mixed theophylline Th and fluorescein-labeled theophylline Th^* compete for a limited number of antibody Ab binding sites in the main mixing and reaction channel that corresponds to channel J3-J4 in Fig. 45. The binding kinetics are governed by the equations in Fig. 47a, where k_1 and k_2 are the forward and backward kinetic constants for the binding reaction between Th and Ab and k_3 and k_4 are those between Th^* and Ab respectively.

The concentrations c of Th , Th^* , Ab , $Ab-Th$ and $Ab-Th^*$ in this simultaneous mixing and binding-reaction are spatial-position dependent and governed by a set of convection-diffusion equations with reactive source terms [99]:

$$\begin{aligned}
u_{Th} \frac{dc_{Th}}{dx} &= D_{Th} \left(\frac{\partial^2 c_{Th}}{\partial x^2} + \frac{\partial^2 c_{Th}}{\partial y^2} \right) - k_1 c_{Th} c_{Ab} + k_2 c_{Ab-Th} \\
u_{Th^*} \frac{dc_{Th^*}}{dx} &= D_{Th^*} \left(\frac{\partial^2 c_{Th^*}}{\partial x^2} + \frac{\partial^2 c_{Th^*}}{\partial y^2} \right) - k_3 c_{Th^*} c_{Ab} + k_4 c_{Ab-Th^*} \\
u_{Ab} \frac{dc_{Ab}}{dx} &= D_{Ab} \left(\frac{\partial^2 c_{Ab}}{\partial x^2} + \frac{\partial^2 c_{Ab}}{\partial y^2} \right) - (k_1 c_{Th} + k_3 c_{Th^*}) c_{Ab} + (k_2 c_{Ab-Th} + k_4 c_{Ab-Th^*}) \\
u_{Ab-Th} \frac{dc_{Ab-Th}}{dx} &= D_{Ab-Th} \left(\frac{\partial^2 c_{Ab-Th}}{\partial x^2} + \frac{\partial^2 c_{Ab-Th}}{\partial y^2} \right) + k_1 c_{Th} c_{Ab} - k_2 c_{Ab-Th} \\
u_{Ab-Th^*} \frac{dc_{Ab-Th^*}}{dx} &= D_{Ab-Th^*} \left(\frac{\partial^2 c_{Ab-Th^*}}{\partial x^2} + \frac{\partial^2 c_{Ab-Th^*}}{\partial y^2} \right) + k_3 c_{Th^*} c_{Ab} - k_4 c_{Ab-Th^*}
\end{aligned} \tag{49}$$

where D is the sample diffusivity, u is the sample velocity and subscripts Th , Th^* , Ab , $Ab-Th$ and $Ab-Th^*$ refer to the quantities associated with the corresponding samples. Eq. (49) is non-linear and does not admit analytical solutions. Ref. [22] suggests several simplifying assumptions:

- The fluorescein has negligible effects on the binding affinity of Th to Ab (namely $k_1 \approx k_3$ and $k_2 \approx k_4$), and on the material properties of Th and Ab-Th (namely $u_{Th} \approx u_{Th^*}$ and $D_{Th} \approx D_{Th^*}$, $u_{Ab-Th} \approx u_{Ab-Th^*}$ and $D_{Ab-Th} \approx D_{Ab-Th^*}$).
- The reaction is essentially irreversible, $k_1 \gg k_2$ and $k_3 \gg k_4$, such that the forward binding is dominant and the analyte exiting mixing-reaction channel does not contain noticeable Ab (the amount of Ab is limited). This can be inferred from the electropherogram (Figure 6) in Ref. [22] (if the reverse reaction was significant, another concentration peak of Th^* disassociated from $Ab-Th^*$ complex would be observed in the separation channel).
- Both mixing and reaction are complete, which are implied by Figures 5 and 7 in Ref. [22] (Figure 5 shows that 99 % mixing has been reached. In Figure 7, except at Th concentration of 10 mg/L, the stop-flow case does not appreciably enhance the reaction, indicating that the reaction is almost complete for the continuous-flow case).

Based on these assumptions, Eq. (49) can be reduced to

$$\begin{aligned}
u_{Th} \frac{dc_{Th}}{dx} &= D_{Th} \left(\frac{\partial^2 c_{Th}}{\partial x^2} + \frac{\partial^2 c_{Th}}{\partial y^2} \right) - k_1 c_{Th} c_{Ab} \\
u_{Th^*} \frac{dc_{Th^*}}{dx} &= D_{Th^*} \left(\frac{\partial^2 c_{Th^*}}{\partial x^2} + \frac{\partial^2 c_{Th^*}}{\partial y^2} \right) - k_1 c_{Th^*} c_{Ab} \\
u_{Ab} \frac{dc_{Ab}}{dx} &= D_{Ab} \left(\frac{\partial^2 c_{Ab}}{\partial x^2} + \frac{\partial^2 c_{Ab}}{\partial y^2} \right) - k_1 (c_{Th} + c_{Th^*}) c_{Ab} \\
u_{Ab-Th} \frac{dc_{Ab-Th}}{dx} &= D_{Ab-Th} \left(\frac{\partial^2 c_{Ab-Th}}{\partial x^2} + \frac{\partial^2 c_{Ab-Th}}{\partial y^2} \right) + k_1 c_{Th} c_{Ab} \\
u_{Ab-Th^*} \frac{dc_{Ab-Th^*}}{dx} &= D_{Ab-Th^*} \left(\frac{\partial^2 c_{Ab-Th^*}}{\partial x^2} + \frac{\partial^2 c_{Ab-Th^*}}{\partial y^2} \right) + k_1 c_{Th^*} c_{Ab}
\end{aligned} \tag{50}$$

The first two equations in Eq. (50) differ from each other by a scaling constant CR , the concentration ratio of c_{Th} to c_{Th^*} , implying that the amount of Ab bound to Th and Th^* (namely, the c_{Ab-Th} and c_{Ab-Th^*}) is proportional to the concentrations of Th and Th^* and CR holds unchanged from the inlet to the outlet of the mixing-reaction channel. Additionally, the assumption of complete mixing and binding-reaction enables the calculation of the concentrations of reaction products and unreacted samples at the end of the mixing-reaction channel (intersection J4) based on the mass-balance principle. In the model implementation, the main mixing-reaction channel in Fig. 47a is modeled as a serial connection of a mixing channel having the same dimensions as the original channel and a reactor with negligible physical size ($R = 0$) in which the binding reaction completes instantaneously (Fig. 47b). This treatment allows the designer to monitor the mixing degree (or mixing residual) before samples and reagents enter the reactor model and examine if the complete mixing assumption is satisfied during design optimizations.

The mixing model is described in the previous chapter and will not be repeated here. The modeling effort hence focuses on the reactor in Fig. 47b. As the amount of Ab is limited (eventually completely consumed), the binding reaction in the reactor model can be quantitatively described by,

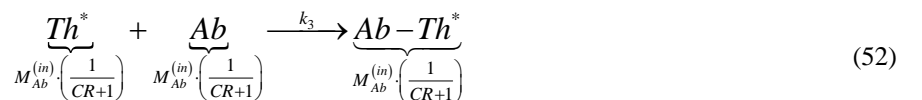
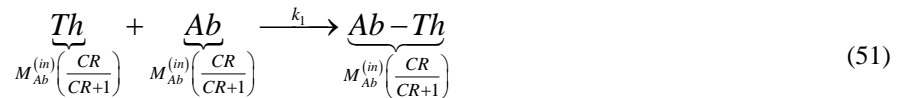


Table 7. Molar mass flow rates of samples and reaction products at the inlet and outlet of the reaction model

| | inlet | outlet |
|--------------------|--|--|
| Th | $M_{Th}^{(in)} = u_{Th} c_{Th}^{(in)} A_r$ | $M_{Th}^{(out)} = u_{Th} c_{Th}^{(in)} A_r - u_{Ab} c_{Ab}^{(in)} A_r \left(\frac{CR}{CR+1} \right)$ |
| Th [*] | $M_{Th^*}^{(in)} = u_{Th^*} c_{Th^*}^{(in)} A_r$ | $M_{Th^*}^{(out)} = u_{Th^*} c_{Th^*}^{(in)} A_r - u_{Ab} c_{Ab}^{(in)} A_r \left(\frac{1}{CR+1} \right)$ |
| Ab | $M_{Ab}^{(in)} = u_{Ab} \cdot c_{Ab}^{(in)} \cdot A_r$ | ~ 0 |
| Ab-Th | 0 | $M_{Ab-Th}^{(out)} = u_{Ab} c_{Ab}^{(in)} A_r \left(\frac{CR}{CR+1} \right)$ |
| Ab-Th [*] | 0 | $M_{Ab-Th^*}^{(out)} = u_{Ab} c_{Ab}^{(in)} A_r \left(\frac{1}{CR+1} \right)$ |

where $M_{Ab}^{(in)} = u_{Ab} \cdot c_{Ab}^{(in)} \cdot A_r$, A_r is the constant cross-sectional area of the mixing-reaction channel, M stands for the molar mass flow rates of samples and indices *in* and *out* represent the quantities at the inlet and outlet of the reactor model. Similarly, those of Th^{*} and Th are expressed as $M_{Th}^{(in)} = u_{Th} \cdot c_{Th}^{(in)} \cdot A_r$, $M_{Th^*}^{(in)} = u_{Th^*} \cdot c_{Th^*}^{(in)} \cdot A_r$. The contribution of the axial diffusive mass flux is not counted due to the long mixing-reaction channel ($L = 81$ mm and $w = 52$ μ m). Then, from mass balance, the molar mass flow rates of the reaction product Ab-Th^{*} and unreacted Th^{*} exiting the reactor model are respectively given by,

$$M_{Ab-Th^*}^{(out)} = u_{Ab} c_{Ab}^{(in)} A_r \left(\frac{1}{CR+1} \right) \quad (53)$$

$$M_{Th^*}^{(out)} = u_{Th^*} c_{Th^*}^{(in)} A_r - u_{Ab} c_{Ab}^{(in)} A_r \left(\frac{1}{CR+1} \right) \quad (54)$$

Table 7 summarizes the molar mass flow rates of samples and reaction products flowing in and out of the reactor.

Denote $M_{Ab-Th^*}^{(out)} = u_{Ab-Th^*} \cdot c_{Ab-Th^*}^{(out)} \cdot A_r$ and $M_{Th^*}^{(out)} = u_{Th^*} \cdot c_{Th^*}^{(out)} \cdot A_r$. The average molar species concentrations of the analytes entering the injector are,

$$c_{avg, Ab-Th^*}^{(out)} = c_{Ab}^{(in)} \left(\frac{1}{CR+1} \right) = d_{0, Ab}^{(in)} \left(\frac{1}{CR+1} \right) \quad (55)$$

$$c_{avg, Th^*}^{(out)} = c_{Th^*}^{(in)} - \frac{u_{Ab}}{u_{Th^*}} c_{Ab}^{(in)} \left(\frac{1}{CR+1} \right) = d_{0, Th^*}^{(in)} - \frac{u_{Ab}}{u_{Th^*}} d_{0, Ab}^{(in)} \left(\frac{1}{CR+1} \right) \quad (56)$$

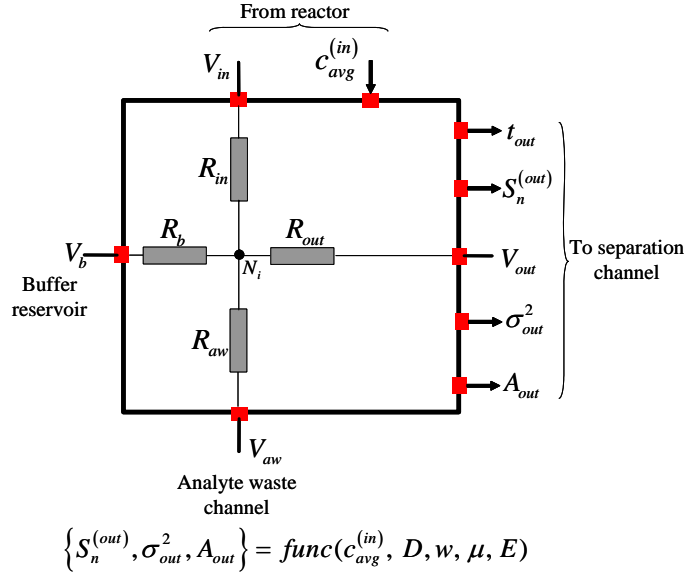


FIGURE 48. Pins for the electrokinetic injector. Indices *in*, *out*, *b*, *aw* refer to the quantities at the terminals respectively linking to the reactor, separation channel, buffer reservoir and sample (analyte)-waste reservoir.

where $u_{Ab} \approx u_{Ab-Th^*}$ is used [22]. The uniform species concentrations (namely the average concentration attributed to complete mixing) $c_{Ab}^{(in)}$, $c_{Th^*}^{(in)}$ and $c_{Th}^{(in)}$ at the inlet of the reactor model can be extracted from the zeroth component of $d_n^{(in)}$. Eqs. (55) and (56) establish the signal flow relationship between the species concentrations at the inlet and outlet of the reactor model.

Obviously, this model does not provide reaction kinetics and requires sufficiently long channels to ensure both complete mixing and reaction (this assumption is valid for the integrated LoC in Fig. 45), leading to conservative assessment of the mixing-reaction channel length for LoC design.

Double-T Injectors. The double-T injector [22] serves as a physical junction between the reactor, separator, sample (analyte)-waste and buffer feed channels. It operates at both loading and dispensing phases and introduces analytes from the continuous-flow reactor into the separation channel that involves transient evolution of species bands. Therefore, it is indispensable in transitioning the LoC from the synthesis to analysis phases. In contrast to the injector model used for separation modeling (Section 4.1.2.), this new model is capable of automatically determining the shape of the injected species band based on the electric fields at both phases and analyte concentrations from the adjacent reactor model.

Pin Definition. Fig. 48 shows the pins at the injector terminals. Since the injector is modeled as a single element, the symbol view does not reflect its real physical geometry, e.g., double-T or cross

[100]. The terminals connecting to the reactor and separation channel are respectively defined as the inlet and outlet, indicating the direction of signal flow for calculating the band shape of the injected species. At the inlet, the information of the average species concentrations ($c_{avg}^{(in)}$) is acquired from the upstream reactor model. At the outlet, the initial species band shape, such as skew coefficients ($S_n^{(out)}$), variance (σ_{out}^2) and amplitude (A_{out}), as well as the starting separation time ($t_{out} = 0$) is set and propagated to the downstream separation channel for separation computation. In addition, four electric pins V_{in} , V_{out} , V_b and V_{aw} are also defined, where indices *in*, *out*, *aw* and *b* represent the quantities at the terminals linking respectively to the reactor, separation channel, sample (analyte)-waste feed channel and buffer feed channel.

Behavioral Model. As its flow path lengths are negligibly small compared with those of feed, mixing-reaction and separation channels, injector can be assumed as a point-wise entity and electrically represented as four resistors with zero resistance between each terminal and the internal node N_i ,

$$R_{aw} = R_b = R_{in} = R_{out} = 0 \quad (57)$$

Here, N_i corresponds to the intersection of flow paths. Thus, the voltages at the terminals are consequently the same ($V_{aw} = V_b = V_{in} = V_{out}$). The functional relationship of the biofluidic states between the inlet and outlet of the injector is determined by electric fields in the adjacent channels and quantitatively described by a parameterized reduced-order model developed with the neural network approach [40].

4.2.3. Connection of Pins

With all behavioral models available, the next step is to link pins at element terminals by wires (electric) and wiring buses (biofluidic) according to the spatial chip layout and compatibility of the physics to accomplish the simulation schematic.

In Table 8, the biofluidic wiring buses are classified into the intra-subsystem (connecting the element within a subsystem, e.g., mixing or separation) and inter-subsystem (linking the subsystems involving different functions) buses. Since the reactor and injector are modeled in terms of a single element, there is no intra-subsystem pin definition.

Table 8. Biofluidic wiring buses used in the integrated LoC simulations

| Connection Type | Subsystem Type | Bus | Pin Name |
|-----------------|----------------------|------------------|--------------------------------|
| Intra-subsystem | Mixing | d [0:29] | Concentration coefficients |
| | | t [0:2] | Separation times |
| | Separation | σ^2 [0:2] | Variance |
| | | A [0:2] | Amplitude |
| | | S [0: 30] | Skew coefficients |
| | Reaction | N/A | N/A |
| Inter-subsystem | Injection | N/A | N/A |
| | Mixing-Reaction | d [0:29] | Concentration coefficients |
| | Reaction-Injection | c_{avg} [0:2] | Average analyte concentrations |
| | Injection-Separation | t [0:2] | Separation times |
| | | σ^2 [0:2] | Variance |
| | | A [0:2] | Amplitude |
| | | S [0: 30] | Skew coefficients |

4.2.4. System Simulation of Integrated LoC Systems

In this section, the system-level schematic simulation procedure will be first described. Then, the simulation results of the integrated competitive immunoassay microchip will be discussed and validated against numerical and experimental data.

Simulation Description. The immunoassay schematic is simulated in two consecutive steps arising from the two operational phases. For each step, both electric and biofluidic simulations are conducted.

- **Mixing-reaction-loading (synthesis) phase.** In this phase, voltage is applied at reservoir 3 with reservoirs 5, 6 and 7 grounded and the others left floating (potential setting is inactivated in the schematic). Given system topology and element dimensions, nodal voltages at element terminals within the entire system are first computed by Ohm's and Kirchhoff's laws using the resistor models. The resulting nodal voltage and current through the element are in turn used to calculate the electric field strength (E) and its direction within the mixer, reactor and analyte-waste feed channel, as well as the flow ratios at intersections J1 and J3. With these results and user-input sample properties (D and μ), the biofluidic pin values are computed sequentially starting from the most upstream sample reservoirs (5, 6 and 7) to the injector. The sample (Th , Th^* and Ab) concentrations after mixing are determined and then fed to the reactor model to calculate the concentrations of detectable species, e.g., the reaction product (Th^*-Ab) and the unreacted Th^* . At the double-T injector, the concentration information along with the electric fields in the adjacent channels is saved.
- **Dispensing-separation-detection (analysis) phase.** In this phase, the voltage is switched to reservoirs 1 and 4 with the others left floating. As the synthesis phase, the nodal voltage in the network is first computed. Then, the computation of separation state (e.g., the arrival time,

Table 9. Comparison between system-level simulation results and experimental data on normalized concentrations of Th^* at channels J1-J3 and J3-J4.

| Channel | System Simulation | Experiment |
|---------|-------------------|------------|
| J1-J3 | 0.5 | 0.534 |
| J3-J4 | 0.25 | 0.257 |

variance, skew and amplitude) is initiated at the injector outlet using the stored information from phase (1), and subsequently propagated to downstream separation channels using signal flow until the waste reservoir is reached.

Simulation Results and Discussion. In this section, the system simulation results and their comparison to experimental and numerical data will be presented to validate the modeling methodology.

Table 9 shows the comparison between the system simulation results and experimental data on the normalized concentrations of Th^* (by the initial reservoir sample concentration) at channels J1-J3 and J3-J4. Since channels J1-reservoir 5 and J1-reservoir 6 have exactly same dimensions (that is same electric resistance), flow rates through them are also identical with a single constant potential applied at reservoirs 5 and 6. This leads to a two-fold dilution of Th^* at channel J1-J3. Likewise, channel J3-J2-reservoir 7 is fabricated with the same resistance as the combined resistance of channels J3-J1-reservoir 5 and J3-J1-reservoir 6, eventually yielding a four-fold dilution of Th^* at channel J3-J4. The simulation results agree with the experimental data excellently with the worst error of 7 %.

In Fig. 49, calibration curves of the area ratio for unreacted Th^* and Ab-Th^* complex are obtained from system simulation results by varying the concentration of Th (Th^* concentration is fixed). Area ratios of Th^* and Ab-Th^* are respectively defined as $\text{Ar}_{\text{Th}^*} / (\text{Ar}_{\text{Th}^*} + \text{Ar}_{\text{Ab-Th}^*})$ and $\text{Ar}_{\text{Ab-Th}^*} / (\text{Ar}_{\text{Th}^*} + \text{Ar}_{\text{Ab-Th}^*})$, where Ar is the species amount, represented by its area under the electropherogram (Fig. 50). The simulation results match the experimental data very well with a relative error less than 5 %. As the concentration of Th increases, the amount of Th begins to predominate in the mixture of Th and Th^* in mixing-reaction channel J3-J4. Therefore, Ab exhibits a higher probability of colliding and binding to Th than to Th^* , leading to more unreacted Th^* and the growth of Th^* area ratio.

Fig. 50 shows the simulated electropherograms of separating Th^* and Ab-Th^* at three detection spots in the separation subsystem. It is seen that species bands of Th^* and Ab-Th^* gradually

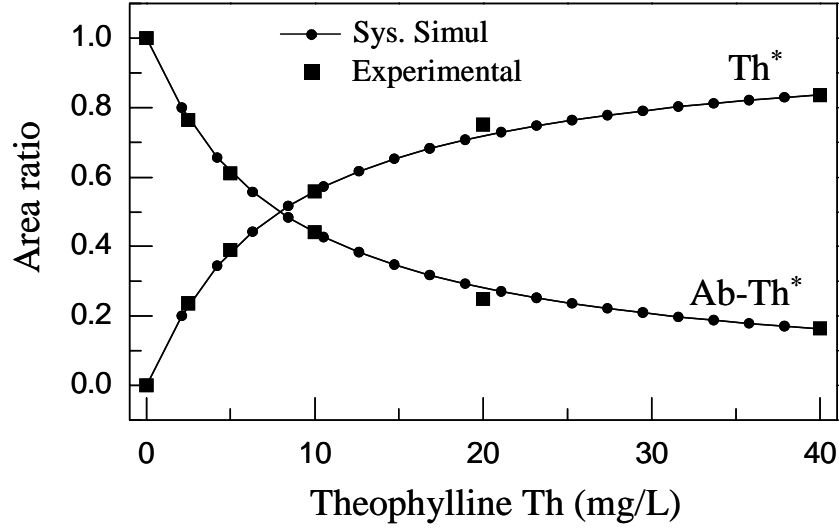


FIGURE 49. Comparison between the system-level simulation results and experimental data [22] on the area ratio of unreacted Th* and Ab-Th* complex. Abscissa shows the initial concentration of Th before the 50-fold off-chip dilution. Actual concentrations of Th* and Ab are not available. Their values were extracted from the results in Ref. [22].

Table 10. Comparison between numerical analysis [29] and system simulations on the area ratio

| | Numerical | Sys. Simul |
|-------------------------------|-----------|------------|
| Area ratio Th [*] | 0.845 | 0.84 |
| Area ratio Ab-Th [*] | 0.155 | 0.16 |

separate during their migration through the separation channel but with a considerable decrease in band's amplitude and spreading in band's width due to the dispersion. The area ratio of the species bands are extracted at the third detector (bottom trace) and compared with numerical simulation [29] in Table 10. Very good agreement with the worst-case error less than 10 % and impressive speedup >100× over the numerical simulations have been achieved.

4.2.5. An Improved Design

Finally, the system-level simulation is employed to redesign the original chip and improve its bio-analysis efficiency and minimize its chip-area. Here, the original constraints on the channel size and power supply are kept. Specifically, the mixing, reaction, injection and separation channels are fabricated with channel width 52 μm and those feed channels leading to buffer and sample (analyte)-waste reservoirs are 236 μm wide. The same power supply with a single output of 6 kV is used. Due to the practical fabrication limit, I/O reservoirs are required to be spaced at least 5 mm apart. In addition, the immunoassay reaction is also assumed to be mixing-limited, thus complete

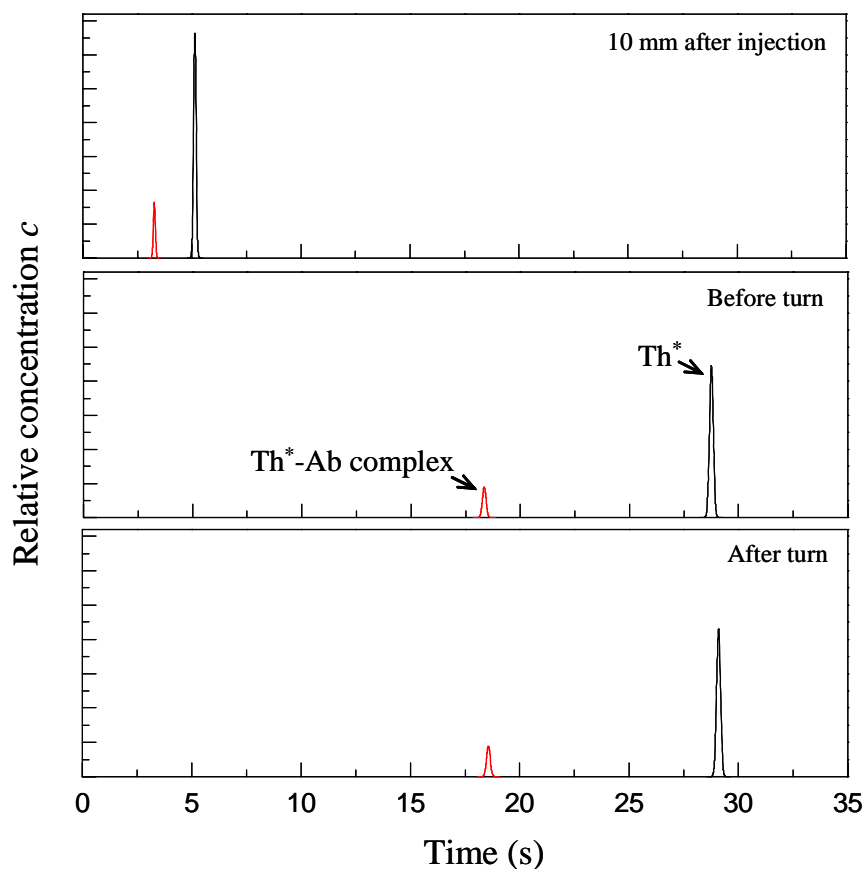


FIGURE 50. Electropherograms of unreacted Th^* and Ab-Th^* complex from system simulations at three detection spots: 10 mm after injection (top trace), before the 90° elbow (middle trace) and after the 90° elbow (bottom trace). Th concentration used in experiments and simulations is 40 mg/L before the 50-fold off-chip dilution (800 $\mu\text{g/L}$ after the dilution).

mixing also signifies complete reaction (otherwise a period of incubation needs to be added at the end of the synthesis phase from the design, in which all reservoirs are left floating).

Fig. 51 shows the redesign by reducing the excessive mixing length and arranging it into a compact serpentine geometry. The original mixing channel J1-J3 and J3-J4 are shrunk from 26.5 and 81.63 mm to 9.3 and 77.33 mm respectively, while keeping almost the same mixing degree. Additionally, the overly long separation channel is shortened from 81.11 to 25.5 mm, which hence increases the electric field (from 600 V/cm to 1.8 kV/cm) and minimizes the band spreading due to diffusion (Joule heating dispersion is still negligible in this case). Furthermore, the detector is moved to the front of the 90° elbow separation channel to avoid the turn dispersion at the high electric field in the new design.

Table 11 shows the modified mixing and separation channel length, as well as the system performance between the original and improved designs. Although the separation resolution drops

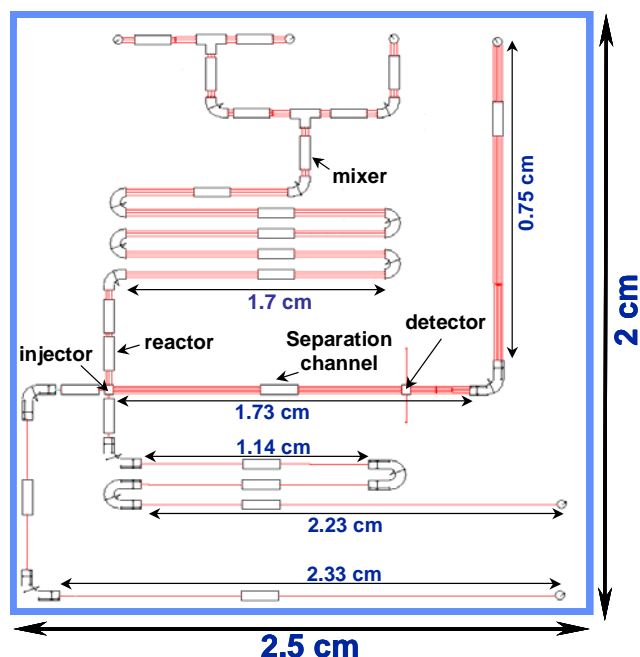


FIGURE 51. Schematic (not to scale) of the improved design of Fig. 45 using the system-level simulation. to 15, it is still high enough to resolve the species bands. Most importantly, an impressive 1.6-fold increase in concentration amplitude and nearly 3-fold and 10-fold decreases in variance and chip area have been achieved. In other words, an assay chip that is easier to detect has been designed in less area.

4.2.6. Summary

This chapter describes the system-level schematic simulation of an electrokinetic competitive immunoassay LoC. In contrast to the separation and mixing models from the previous chapters,

Table 11. Comparison of the channel dimension, separation time, variance, peak height and chip area between the original design and the improved design.

| Parameters | Original | | Improved | |
|------------------------------|------------------------|-----------------|----------------------|-----------------|
| J1-J3 | 26.5 mm | | 9.3 mm | |
| J3-J4 | 81.6 mm | | 77.33 mm | |
| Mixing degree | 99 % | | 99 % | |
| J4-reservoir 4 | 81.11 mm | | 25.5 mm | |
| | Ab-Th [*] | Th [*] | Ab-Th [*] | Th [*] |
| Separation Time (s) | 18.6 | 29.1 | 2.12 | 3.32 |
| Variance (μm^2) | 66607 | 30091 | 15926 | 11400 |
| Amplitude (norm.) | 1 | 4.9 | 2.4 | 8 |
| Resolution | 24 | | 15 | |
| Chip area | 7.6 cm \times 7.6 cm | | 2.5 cm \times 2 cm | |

pins involving different physics have been defined at the terminals of reactor and injector models, which enable the interconnection among different subsystems and transition between the operational phases of the chip. A simplified reactor model valid for forward immunoassay reactions has been developed, implemented in an analog hardware description language (Verilog-A) and linked to behavioral models of mixers, separation channels and injectors [40] to form a complete simulation schematic. An overview of the pin connections within and between subsystems has been provided to interpret the biofluidic signal transmission within the entire LoC network.

The constructed schematic then has been used to simulate the LoC in both synthesis and analysis phases, each requiring sequential solutions of the electric network by Kirchhoff's law and biofluidic states by signal flow. The simulation results of on-chip mixing, reaction, injection and separation have been verified by numerical and experimental data. A speedup ($>100\times$) is achieved over the numerical FEM simulations, while still maintaining high accuracy (relative error less than 10 %). The system-level simulation captures the overall effects of chip topology, element size and operational parameters on chip performance and is used to redesign the original chip to improve analysis quality but occupy less chip-area. This application of the modeling and simulation framework developed in this thesis demonstrates its effectiveness in system-level design of integrated LoCs.

4.3. Component Optimization

Our design approach has three fundamental components: (1) The system-level simulator described in the preceding sub-sections, which is capable of simulating complex designs in seconds (2) General area constraints and heuristic *Channel Packing and Placement* algorithms (CPP). (3) Tailored design optimization methods that utilize the system simulator, CPP algorithms, numerical optimization and area constraints.

4.3.1. Design Constraints

We have approached the design of micro electrophoretic devices from two perspectives. Specifically we wish to: (1) Determine the highest performance design that can fit within a given chip area. (2) Minimize the area occupied by a design while maintaining device feasibility. The trade-offs between these two objectives will be discussed in this report.

We have divided the constraints on the design problem into *performance constraints*, *operating constraints* and *physical constraints* as summarized in Table 12.

Table 12.Examples of three different constraint classes along with typical numerical values.

| Performance | Operating | Physical |
|------------------------|--------------------------|--|
| $N \geq 10^4$ | $C \geq 1\mu\text{M}$ | $\omega \in [10,100]$ μm |
| $H \leq 10 \text{ mm}$ | $E \leq 10 \text{ V/cm}$ | $\text{Area} \leq 25 \text{ cm}^2$ |
| $R \geq 1.5$ | $V \leq 30 \text{ kV}$ | $\text{PAD} \geq 5 \mu\text{m}$ |

Performance constraints provide a lower bound on the effectiveness of the separation system. In our case we require that separations always achieve baseline resolution (≥ 1.5). Operational constraints are dictated by external equipment such as the available voltage source or the detector sensitivity. Even if this equipment does not occupy physical space on a chip, it can have a significant impact on the final design. Physical constraints are those dictated by regions of model applicability and fabrication limits. Fabrication constraints capture manufacturing limits; for instance, the minimum channel width ω or the minimum spacing between channels PAD .

Currently, since the range of possible channel widths ω and channel depths δ is small, we treat these quantities as parameters and fix their values. In general, we find that smaller channel widths provide better performance and smaller area. Furthermore, our models can neglect the boundary layer effects that become important in very small channels. Our smallest channel dimension is well over an order of magnitude larger than the Debye length [36] produced by a standard CE electrolyte in a glass or plastic chip.

Connectivity and Area Constraints. The two most important physical constraints are what we call connectivity constraints and area constraints. Connectivity constraints are used to uniquely specify a particular channel topology. Area constraints are used to determine the space requirement for a given channel topology on the chip.

Fig. 52 illustrates the three parts of our connectivity constraints by showing the information necessary to connect a U-bend to a straight channel section. It is apparent that the U-bend must attach to the end of the straight. However, this results in four possible scenarios as shown in Fig. 52(a). Two of these turns are illegal (turns 3 and 4) because they double back on the straight section. Doubling back can be eliminated by enforcing a flow direction as shown in Fig. 52(b). The flow direction is a cardinal direction $\{\text{north}, \text{south}, \text{east}, \text{west}\}$ where the flow direction out of an upstream section must equal the flow direction in to the downstream section. This leaves turns 1 and 2 as viable alternatives. The topology can be uniquely specified by providing a rotational direc-

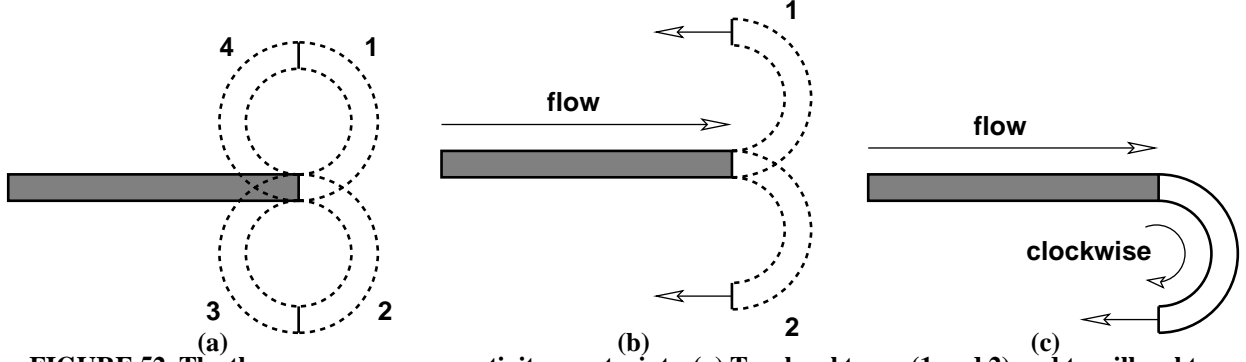


FIGURE 52. The three necessary connectivity constraints. (a) Two legal turns (1 and 2) and two illegal turns (3 and 4) that can be attached to the endpoint of a straight channel. (b) The outlet flow direction of the straight and the inlet flow direction of the turn must match, thereby eliminating illegal turns. (c) The topology becomes unique when a turn direction is specified.

tion $\{clockwise, counterclockwise\}$ to the turn as shown in Fig. 52(c). These three pieces of information allow a designer to build a unique 2 section channel topology from the library of section types discussed in Section 4.2.

However, the amount of information required to uniquely specify any particular topology grows with the number of channel sections. Fig. 53 shows three of the possible scenarios that may occur if a second clockwise turn is added to the topology shown in Fig. 52(c). Fig. 53(a) is illegal because it overlaps the initial straight section. Fig. 53(b) is a legal topology where the initial straight section has been shortened. Fig. 53(c) is a legal topology where the first clockwise turn has been lengthened. As sections are added, the problem becomes more combinatorial. We eliminate this complication by focusing on standardized serpentine and spiral topologies, which are both highly compact and readily integrability with other on-chip components.

The definition of a serpentine or spiral channel topology can be contained in an object \mathcal{J} . Serpentine topologies can be represented as $\mathcal{J} = [L^{str}_1, L^{trn}_2, L^{str}_3, L^{trn}_4, \dots, L^{str}_n, L^{trn}_m]$. The odd ele-

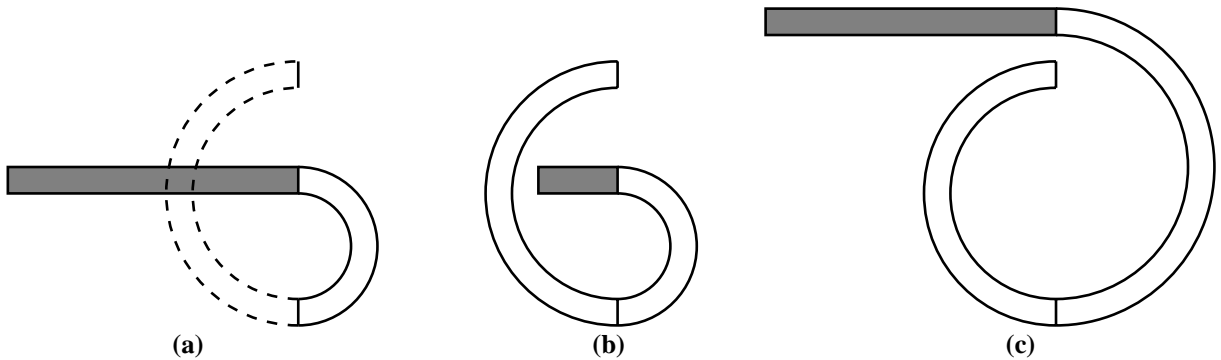


FIGURE 53. Adding a second turn. (a) An illegal clockwise turn (b) Legalize by shortening the initial straight section. (c) Legalize by lengthening the first clockwise turn.

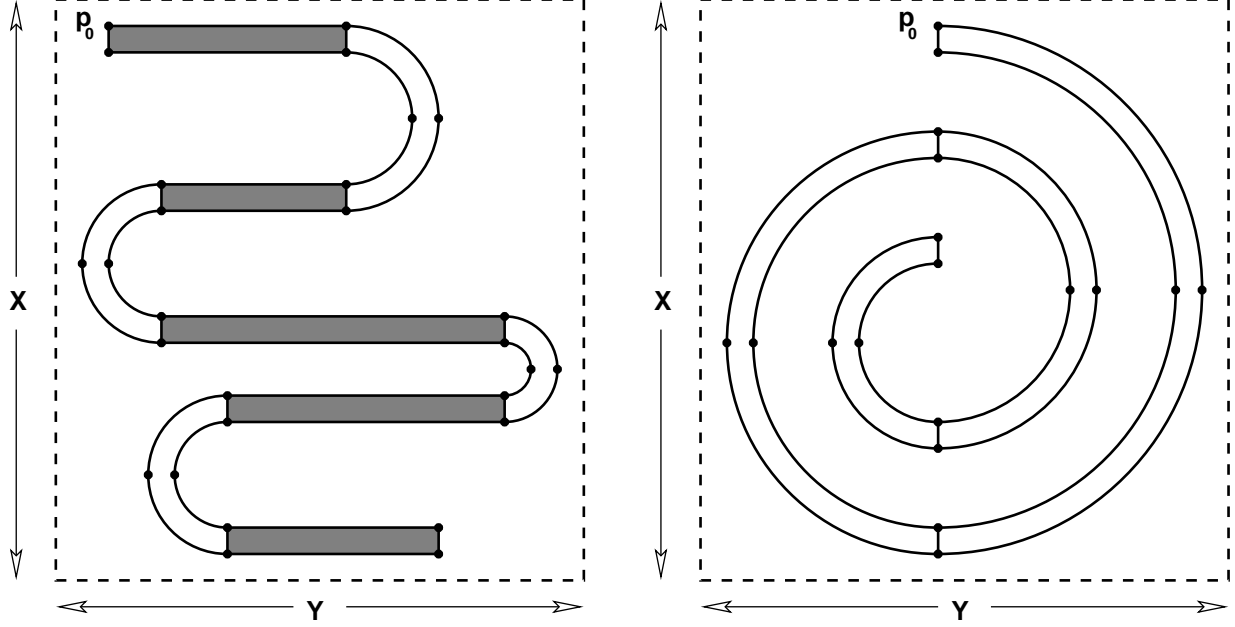


FIGURE 54. Bounding-box determination for (a) serpentine topologies and (b) spiral topologies. All points are calculated with respect to P_0 (the origin). The X and Y dimensions are the actual maximum topology dimensions plus $2PAD$, where PAD is a minimum feature size specification.

ments of \mathcal{J} are straight section lengths L^{str}_n where $n = 1, 3, 5 \dots |\mathcal{J}|-1$, and the even elements are turn lengths L^{trn}_m along the center-line radius of the channel, where $m = 2, 4, 6 \dots |\mathcal{J}|$. Spiral topologies can be represented as an array $\mathcal{J} = [L^{trn}_1, L^{trn}_2, \dots, L^{trn}_m]$. The length of each U-bend L^{trn}_m is measured along the center line radius of the channel and $m = 1 \dots |\mathcal{J}|$.

Once a topology has been specified, the space required to fit it on a chip can be determined by measuring the area or bounding-box of the topology. Fig. 54 shows how the bounding box of a serpentine and spiral topology is determined. A set of points P along the edges of the topology are determined starting from point $p_0 = (0,0)$, $p_0 \in P$. All subsequent points are tracked with respect to p_0 .

The width X and height Y of a topology can be readily determined according to Eqs. (58) and (59).

$$X = \max(x_i, \forall x_i \in P) - \min(x_i, \forall x_i \in P) \quad (58)$$

$$Y = \max(y_i, \forall y_i \in P) - \min(y_i, \forall y_i \in P) \quad (59)$$

The simulator described in Section 4.2 as well as the connectivity and area constraints are used to evaluate each candidate design as shown in Eq. (60). This function, labelled ASIM, is capa-

ble of simulating and calculating the bounding box dimensions of any arbitrary topology.

$$[X, Y, E, R_{ij}, C_i] = ASIM(V, geom, props, buffer) \quad (60)$$

The simulator takes in the voltage drop V along the channel, and the topology object $geom$ where the instance $\mathcal{I} \in geom$. The $props$ object which contains analyte physiochemical properties and the $buffer$ object which contains the buffer physiochemical properties are also passed into the simulator. The bounding box dimensions, X and Y , are calculated internally and returned along with the electric field strength E as well as the separation resolution $R_{i,j}$ and peak concentrations C_i for each band at the end of the channel. ASIM forms the core of the NLP formulations that we present in the following section because it maps the design variables (channel section lengths and voltage) to device size and performance.

4.3.2. Non-Linear Programming Optimization

Here we describe non-linear programming (NLP) optimization approaches and problem formulations to directly optimize the serpentine and spiral topologies [101]. Our approach is to repeatedly solve optimization problems for each topology, incrementing the number of sections in the topology up to a value NS , which is the maximum number of channel sections that can fit within the bounding box defined by X_{max} and Y_{max} . NS is readily found using a channel placement and packing heuristic [102]. Optimizing each NLP individually is effective since the problem is not combinatorial.

Our approach results in NS locally optimal solutions describing how topology size changes with the number of sections in the topology. There are several methods available that can be used to solve the serpentine and spiral optimization problems [103]. We were able to apply a standard successive quadratic programming (SQP) [43] algorithm to solve the spiral optimization problem. However, a more tailored approach was required to optimize the serpentine because our objective function is non-smooth, which results in discontinuous gradients. The non-smoothness results from the fact that both increasing and decreasing the length of an inter-turn straight section can potentially result in an increase in design area. Furthermore, the problem is non-convex and the constraints can become difficult to satisfy when certain physical property values or bound specifications are used.

Adaptive Penalty Parameters. A standard penalty function method with fixed penalties must be minimized several times to satisfy the constraints of the original NLP to a high level of accuracy. A

common way to deal with this inefficiency is to minimize an *exact penalty function* where the L_1 norm of the constraint violation is added to the objective function. If a sufficiently large penalty parameter is chosen, the solution to the original NLP and the penalized problem correspond exactly; however, the penalized problem is non-smooth. We found that in our work, the exact penalty formulation would often fail to converge. We speculate that this occurred because the large penalty parameters required for constraint satisfaction also introduce ill conditioning in the Hessian. Also, the large number of constraints in our problem introduce a great deal of non-smoothness into the problem.

Instead, we have chosen to address this problem by changing the scalar penalty parameter p^k into an adaptive penalty parameter p_i^k as shown in Eq. (61) for each of the constraints $i = 1 \dots m$ in the original NLP. p_i^k is calculated based on the scaled constraint violation from the previous iteration $k-1$.

$$\begin{aligned} \min F(x^k, p^k) &= X \cdot Y + \sum_{i=1 \dots m} p_i^k \left(\frac{\max(0, g_i(x^k))^2}{\text{typ}X} \right) \\ p_i^k &= \rho^k \left(\frac{\max(0, g_i(x^{k-1}))}{\text{typ}X} \right) \quad \forall i = 1 \dots m \end{aligned} \tag{61}$$

In Eq. (61), $x^k = \{J, V\}$ for the current iteration, ρ is an increasing sequence 10, 100, 1000, ..., and $\text{typ}X$ is a typical x value meant to scale $g_i(x)$ so that all constraint residuals are of roughly the same order.

This is an intuitive approach based on the fact that the Lagrange multiplier $\lambda_i^k \approx p_i^k \cdot \max(0, g(x^k))$ and it approaches its optimal value λ_i^* as F and x approach their optimal values F^* and x^* . The scalar parameter ρ^k is necessary to ensure that $p_i^k \geq \lambda_i^k$ so that each constraint is appropriately enforced.

The procedure begins by obtaining p_i^1 from the initial point x^0 . Subsequent values of p_i^k are calculated based on the solution to the previous iteration x^{k-1} . Our computational experience indicates that our adaptive penalty parameter approach typically requires 1 to 2 fewer sequential minimizations than the standard approach.

Table 13. Spiral Design Data (extracted from [20]).

| Data | Values |
|--------------------------------|--|
| Actual Design Area | $3.7 \text{ cm} \times 3.8 \text{ cm}$ ($A = 14 \text{ cm}^2$) |
| Total Separation Length | 22.2 cm |
| Channel Width | 40 μm |
| Channel Depth | 15 μm |
| Applied Electric Field | 1170 V cm^{-1} |
| Applied Voltage (over 22.2 cm) | 26.0 kV |
| Maximum Available Voltage | 30.0 kV |
| NUmber of Plates (@22.2 cm) | $N_{DCF} = 1.04 \times 10^6$ |

4.3.3. Example Case Studies

Here we compare designs generated using our techniques to a serpentine design developed by Jacobson et al. [62] and a spiral design developed by Culbertson et al. [20]. We attempted to include industrially relevant fabrication specifications. An minimum feature size specification of $PAD \geq 5 \mu\text{m}$ was obtained from a MEMS and microfluidics foundry [104]. Using our methods, we were able to reduce the size of the literature designs while still maintaining the desired level of separation performance.

Optimization of a Spiral Design. Culbertson et al. present a spiral design [20] with 4 channel sections between the injector and detector. The entire design, including all wells, fits within a $5\text{cm} \times 5\text{cm}$ area. The spiral topology itself fits within approximately a $3.7\text{cm} \times 3.8\text{cm}$ area. In our comparison we will focus only on the area occupied by the spiral topology. The chemical species used was dichlorofluoroscein (DCF) and its physical properties were extracted from results presented by Culbertson et al. The buffer used was a boric acid/TRIS solution. Table 13 lists some of the important design features and data.

Fig. 55(a) shows a schematic of the separation portion of the original spiral topology. Here we use the optimization methods described in section Section 4.3.2 to design a spiral separation system that is smaller than the original design and maintains the same or greater plate number ($N_{DCF} \geq 1.04 \times 10^6$).

Fig. 55(a), shows the original design, which is essentially 4 semicircles with large radii and narrow channel widths. This configuration leads to low dispersion. However, the design leaves a great deal of inter-channel spacing which allows us to significantly compact the design. In

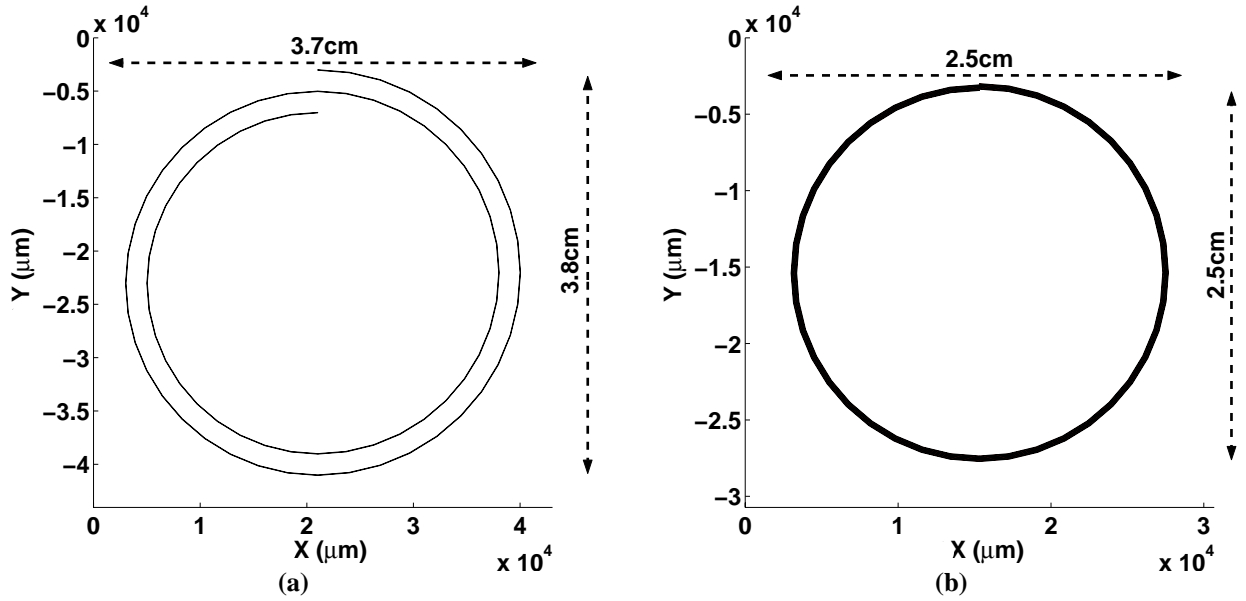


FIGURE 55. Improvement of a spiral design: (a) A schematic of the design presented in [20] (b) A schematic of an improved design generated using our optimization algorithm.

Fig. 55(b) we see that the design can actually be reduced in area by approximately 56 %.

This results from the addition of 6 new channel sections resulting in a 10 section design. An applied voltage of 27 kV was used, which leaves an extra 3 kV to connect the separation channel to the wells.

The channels are compressed as closely as possible to each other without violating the constraint on $PAD \geq 5 \mu m$. Fig. 55 shows a magnification of the optimized design. It can be seen that the feature spacing requirement is met. Furthermore, this design retains approximately 4.5 cm^2 of

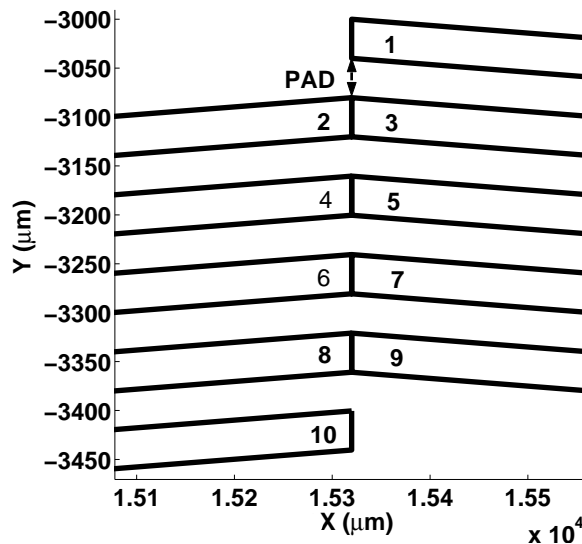


FIGURE 56. Magnification of the optimized spiral design shown in Fig. 55.

space interior to the spiral for the placement of the waste well. This design could be further compressed by either adding more sections, increasing the applied voltage or decreasing the required inter-channel spacing.

4.3.4. Summary

In this section we formulated the layout optimization problem for serpentine and spiral topologies. We investigated numerical techniques to directly solve our NLP formulations. We found that while a conventional SQP algorithm is effective for spiral topologies, other methods must be applied for serpentine topologies.

4.4. Place and Route

Multiplexed LoCs have many potential areas of application, especially in the biomedical industry, where device size is highly constrained. For instance, in-vivo diagnostic chips that are implanted into living tissue must be both small and contain a high degree of functionality. Hand-held point-of-care (PoC) medical devices are another important example. In these devices, size is not only a major constraint, but also the primary cost consideration, since LoCs for PoC applications are often disposable.

Currently, the most complex multiplexed LoCs consist of arrays, where hundreds of replicated simple channel structures function in parallel [3]. The typical layout of these devices is in a spoked-wheel configuration with straight separation channel sections making up the spokes. Each subsystem design is defined by the set of chemical properties and design specifications discussed in Section 4.3.

4.4.1. Floorplan Problem Definition

We define our floor plan problem as follows [105]: given a set of subsystems \mathcal{S} and their associated input and output wells \mathcal{W} obtain a planar arrangement A of subsystems and wells that does not exceed the total chip height H or width W , and where all subsystems with respect to the constraint set in the serpentine optimization problem are feasible. The objective is to choose the arrangement of subsystems and wells that is most compact and where fluid I/O wells are positioned on the edge closest to their associated subsystem ports.

We divide the formulation into 5 parts: P1 - Subsystem rotation, P2 - Subsystem overlap, P3 - Well placement, P4 - Well to port assignment, and P5 - Boundary constraints and objective.

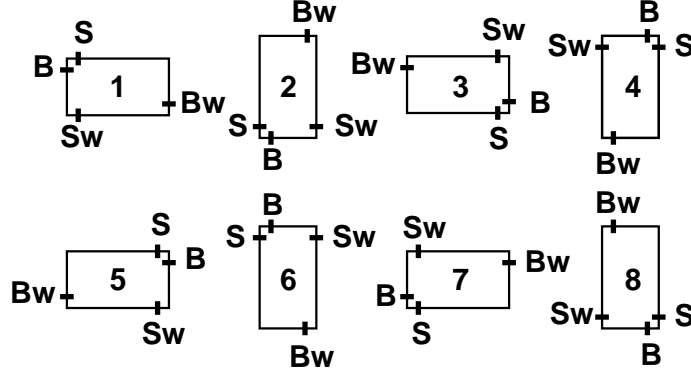


FIGURE 57. Eight possible subsystem orientations.

Subsystem Orientation (P1). In Eq. (62) we define the 8 possible orientations and associated port locations for each subsystem i in \aleph . The 8 orientations are rotations and reflections of a serpentine subsystem as shown in Fig. 57.

$$\forall r = 1 \dots 8 \quad \vee \quad \begin{bmatrix} Z_i^r \\ S_i = (S^x, S^y) \\ Sw_i = (Sw^x, Sw^y) \\ B_i = (B^x, B^y) \\ Bw_i = (Bw^x, Bw^y) \end{bmatrix} \quad \forall i \in \aleph \quad (62)$$

In Eq. (62), one of the Boolean variables, Z_i^1 through Z_i^8 will be assigned *True* depending on its orientation as shown in Fig. 57. When Boolean variables are translated into mixed integer form, they are assigned a binary value, *True* = 1 or *False* = 0.

The coordinates of the sample (S_i), sample waste (Sw_i), buffer (B_i), and buffer waste (Bw_i) ports are determined for an orientation of a subsystem with respect to the bottom left-hand corner of a particular block. For example, the buffer port location B_i when $Z_i^4 = \text{True}$ is $B_i = (B_x, B_y) = (x_i + Y_i - \omega - PAD, y_i + X_i)$. The port locations are a function of the subsystem's location, x_i and y_i , length X_i and height Y_i , as well as the minimum feature spacing PAD and channel width ω .

Overlap Prevention (P2). Since we assume a building block layout style, blocks can be placed anywhere on the chip, but a legal placement is only achieved when no blocks overlap. Overlap is

prevented by assigning $\{right, left, above, below\}$ relationships between each block in \aleph , ($i, j \in \aleph$). Here we show a GDP that enforces these relationships for i left of j (Eq. (63)), i right of j (Eq. (64)), i below j (Eq. (65)) and i above j (Eq. (66)).

$$\left[\begin{array}{c} \delta_{i,j}^1 \\ \bigvee_{\forall r = 1, 3, 5, 7} \left[\begin{array}{c} Z_i^r \\ x_i + X_i \leq x_j \end{array} \right] \bigvee_{\forall r = 2, 4, 6, 8} \left[\begin{array}{c} Z_i^r \\ x_i + Y_i \leq x_j \end{array} \right] \end{array} \right] \bigvee \quad (63)$$

$$\left[\begin{array}{c} \delta_{i,j}^2 \\ \bigvee_{\forall r = 1, 3, 5, 7} \left[\begin{array}{c} Z_i^r \\ x_i - X_i \geq x_j \end{array} \right] \bigvee_{\forall r = 2, 4, 6, 8} \left[\begin{array}{c} Z_i^r \\ x_i - Y_i \geq x_j \end{array} \right] \end{array} \right] \bigvee \quad (64)$$

$$\left[\begin{array}{c} \delta_{i,j}^3 \\ \bigvee_{\forall r = 1, 3, 5, 7} \left[\begin{array}{c} Z_i^r \\ y_i + Y_i \leq y_j \end{array} \right] \bigvee_{\forall r = 2, 4, 6, 8} \left[\begin{array}{c} Z_i^r \\ y_i + X_i \leq y_j \end{array} \right] \end{array} \right] \bigvee \quad (65)$$

$$\left[\begin{array}{c} \delta_{i,j}^4 \\ \bigvee_{\forall r = 1, 3, 5, 7} \left[\begin{array}{c} Z_i^r \\ y_i + Y_i \geq y_j \end{array} \right] \bigvee_{\forall r = 2, 4, 6, 8} \left[\begin{array}{c} Z_i^r \\ y_i + X_i \geq y_j \end{array} \right] \end{array} \right] \bigvee \quad (66)$$

$$ord(i) < ord(j), \forall i, j \in \aleph$$

The values of the binary variables $\delta_{i,j}^1$, $\delta_{i,j}^2$, $\delta_{i,j}^3$, and $\delta_{i,j}^4$ determine the orthogonal packing of the blocks. A disjunction is active when its associated binary equals one. The disjunctions over Z_i^r incorporate the block orientation information into the block packing problem.

Well Placement (P3). We require that fluid I/O wells are placed on the chip's periphery as shown in Fig. 58. Eq. (67) shows the nested disjunctions necessary to model the placement of a well on the top edge of the chip. We use the variables ϕ_i^t , ϕ_i^r , ϕ_i^b and ϕ_i^l , to determine if well $i \in \mathcal{W}$ is placed on the top, right, bottom or left edge of the chip respectively. The value of ϕ_{ij} dictates the relative posi-

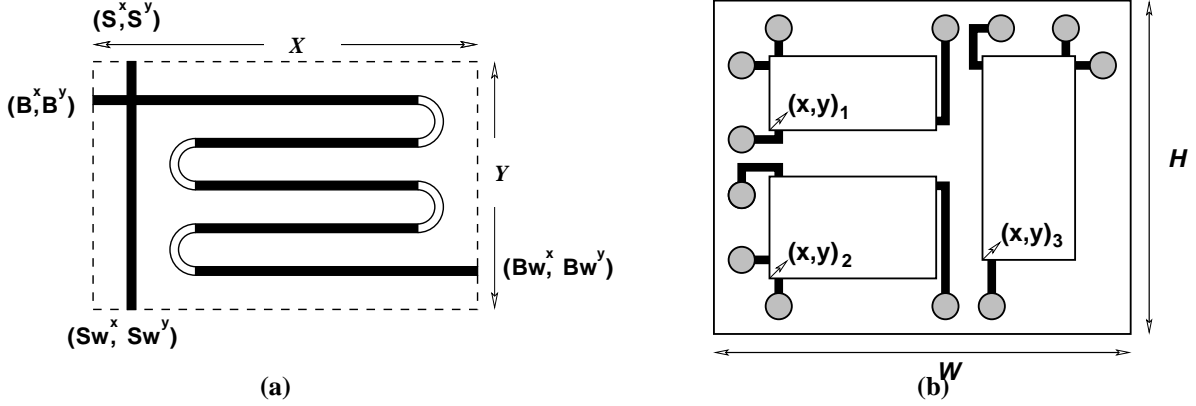


FIGURE 58. (a) Schematic of a serpentine subsystem showing horizontal X and vertical Y dimensions, and port locations (S^x, S^y) , (B^x, B^y) , (Sw^x, Sw^y) , and (Bw^x, Bw^y) . (b) Three subsystem chip layout showing the horizontal W and vertical H dimensions and subsystem locations $(x, y)_i$. Fluid I/O wells are shown as circles, and auxiliary channels (black lines) connect wells and subsystems.

tion of well i with respect to well $j \in \mathcal{W}$ if it exists on that particular edge. A well's position is defined by a coordinate point in the center of the well (W_k^x, W_k^y) . Its placement is a function of the the chip's width W and height H , as well as the well's diameter d and the PAD around every well. The disjunctions for the right, bottom, and left edges are similar to Eq. (67).

$$\begin{aligned}
 & \left[\begin{array}{c} \phi_i^t \\ W_i^y = H - PAD - \frac{1}{2}d \\ \vee \\ \forall \{t, r, b, l\} \left[\begin{array}{c} \phi_j^t \\ \left[\begin{array}{c} \phi_{i,j} \\ W_i^x \leq W_j^x - PAD - d \\ W_i^x \geq PAD + \frac{1}{2}d \end{array} \right] \vee \left[\begin{array}{c} \neg \phi_{i,j} \\ W_i^x \geq W_i^x + PAD + d \\ W_i^x \leq W - PAD - \frac{1}{2}d \end{array} \right] \\ \left[\begin{array}{c} \neg \phi_j^t \\ W_i^x \geq PAD + \frac{1}{2}d \\ W_i^x \leq W - PAD - \frac{1}{2}d \end{array} \right] \end{array} \right] \vee \end{array} \right] \quad (67)
 \end{aligned}$$

The inner-most disjunction in Eq. (67) could be eliminated by constraining the Euclidean or rectilinear distance between wells i and j ; however, this reformulation would introduce more non-

linearity to the system.

Well to Port Assignment (P4). Here we model the connection between subsystem ports and wells as an assignment problem. We require that there is a unique one to one mapping between ports and wells. Our goal is to minimize the distance that auxiliary channels must span to connect a port to a well. Eqs. (68) enforce that a port of a particular subsystem is assigned to only one well.

$$\begin{aligned} \sum_{j \in W} P_{i,j}^S &= 1 & \sum_{j \in W} P_{i,j}^{Sw} &= 1 \\ \sum_{j \in W} P_{i,j}^B &= 1 & \sum_{j \in W} P_{i,j}^{Bw} &= 1 \end{aligned} \quad \forall i \in \aleph \quad (68)$$

The binary variables, $P_{i,j}^S$, $P_{i,j}^{Sw}$, $P_{i,j}^B$ and $P_{i,j}^{Bw}$ represent the connection of the sample, sample waste, buffer and buffer waste ports to a particular well. Eqs. (69) enforces that a well is connected to a single port.

$$\sum_{i \in \aleph} (P_{i,j}^S + P_{i,j}^{Sw} + P_{i,j}^B + P_{i,j}^{Bw}) = 1 \quad \forall j \in W \quad (69)$$

We use a rectilinear distance metric [106] to estimate the length that an auxiliary channel must span between a port $i \in \aleph$ and a well $j \in W$. The connection length between a particular port and well represents the cost of an assignment and is captured by the variables $C_{i,j}^S$, $C_{i,j}^{Sw}$, $C_{i,j}^B$ and $C_{i,j}^{Bw}$ as shown in Eq. (70). Notice that the cost of an assignment is not fixed as it is in typical assignment problems, but is a function of the chip's size and the size and position of each subsystem.

$$\begin{aligned} C_{i,j}^S &= |S_i^x - W_j^x| + |S_i^y - W_j^y| \\ C_{i,j}^{Sw} &= |Sw_i^x - W_j^x| + |Sw_i^y - W_j^y| \\ C_{i,j}^B &= |B_i^x - W_j^x| + |B_i^y - W_j^y| \\ C_{i,j}^{Bw} &= |Bw_i^x - W_j^x| + |Bw_i^y - W_j^y| \end{aligned} \quad (70)$$

In practice, Eqs. (70) are implemented as a system of linear inequalities which avoids the nonlinearity introduced by the $abs(\bullet)$ operator.

Boundary Constraints and Objective (P5). The bounding box of the design is calculated in Eqs. (71), where W and H are the calculated design width and height respectively. Notice that all sub-

systems and wells are taken into account.

$$\begin{aligned}
x_i + X_i \sum_{r=1}^4 Z_i^{2r-1} + Y_i \sum_{r=1}^4 Z_i^{2r} + \phi_i^r(d + 2PAD) + \phi_i^l(d + 2PAD) &\leq W \\
y_i + Y_i \sum_{r=1}^4 Z_i^{2r-1} + X_i \sum_{r=1}^4 Z_i^{2r} + \phi_i^t(d + 2PAD) + \phi_i^b(d + 2PAD) &\leq W
\end{aligned} \tag{71}$$

When the available chip width \bar{W} and height \bar{H} are specified, W and H must be constrained such that $W < \bar{W}$ and $H < \bar{H}$. However, these constraints add a great deal of computational burden to an already difficult problem.

Here we have chosen to use the weighted sum of the bounding box perimeter and the total auxiliary channel length as our objective function (Eq. (72)). The weights, α_1 and α_2 , are assigned values to emphasize either the area minimization or the auxiliary channel length minimization. However, choosing good values for α_1 and α_2 is often difficult. We will discuss our solution to this issue later in the report.

$$\min F = \alpha_1(W + H) + \alpha_2 \sum_{i \in \mathbb{N}} \sum_{j \in W} (C_{i,j}^S P_{i,j}^S + C_{i,j}^{Sw} P_{i,j}^{Sw} + C_{i,j}^B P_{i,j}^B + C_{i,j}^{Bw} P_{i,j}^{Bw}) \tag{72}$$

Problems P1 - P5 represent a rigorous yet highly coupled, highly combinatorial nonlinear description of the multiplexed LoC design and layout problem. This kind of formulation is typically translated into a mixed integer nonlinear program (MINLP) and solved using deterministic optimization approaches that combine gradient-based search methods with Branch and Bound [107]. However, the floor planning problem that we have formulated is a rectangle packing problem that has been shown to be NP-hard [108]. In addition, floor plan optimization using Branch and Bound [109] or conventional mixed integer linear programming (MILP) approaches [110] often have difficulty handling problems of realistic size. Since we are coupling rectangle packing with a non-convex (nonlinear) physiochemical problem, our problem is more difficult than rectangle packing alone. In fact, even if we neglect parts P1, P3 and P4 of the problem, we can only solve for trivial instances (i.e. $\mathbb{N} \leq 4$ and $W = \emptyset$) using conventional MINLP solvers.

4.4.2. Combinatorial Floorplan Problem Formulation

We now formulate the multiplexed LoC floorplanning problem as a combinatorial optimiza-

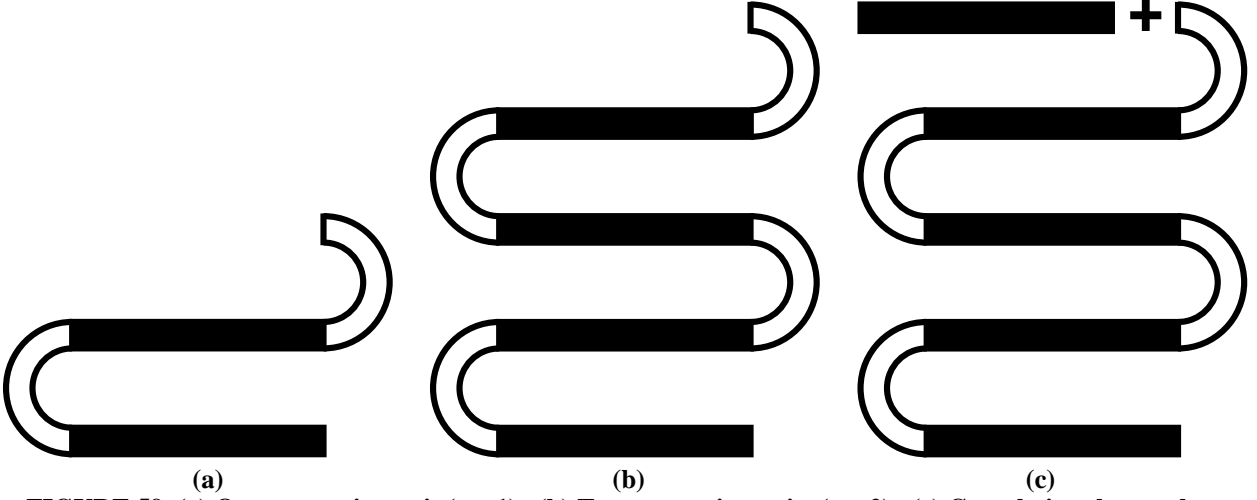


FIGURE 59. (a) One serpentine unit ($\tau = 1$). (b) Two serpentine units ($\tau = 2$). (c) Completing the topology with an initial straight section.

tion problem directly (the above GDP approach clearly requires a heuristic for solution).

Subsystem Orientation. As discussed earlier, each subsystem can have one of eight possible orientations (Fig. 57). Different orientations contribute to design compaction. We define a vector $\vec{Z} = (z_1, z_2, \dots, z_{|\mathbb{S}|})$ which contains the subsystem orientation labels $z_i \in \{1, \dots, 8\}$ where i is a unique element of \mathbb{S} . Each z_i defines the orientation of subsystem i and allows for the deterministic calculation of that subsystem's port locations. For example, the buffer port location B_i for orientation $z_i = 4$ is $B_i = (B^x, B^y) = (x_i + w_i - \omega - PAD, y_i + h_i)$, which corresponds to the case when $z_i^4 = True$ in the GDP. The port locations are a function of the subsystem's location x_i and y_i , width w_i and height h_i , as well as the feature spacing PAD and channel width ω . However, recall that the performance simulation has no knowledge of subsystem orientation. Thus, Eq. (73) must be applied to translate the general subsystem dimensions, X_i and Y_i , used in the simulator to the actual subsystem dimensions w_i and h_i .

$$w_i = \begin{cases} X_i & \text{if } z_i \text{ odd} \\ Y_i & \text{otherwise.} \end{cases} \quad h_i = \begin{cases} X_i & \text{if } z_i \text{ even} \\ Y_i & \text{otherwise.} \end{cases} \quad (73)$$

Serpentine Topology Size. The types list \mathcal{J} can be reduced to a scalar value τ indicating the number of serpentine units within a subsystem topology as shown in Fig. 59. Fig. 59(a) and (b) show a topology with $\tau = 1$ and $\tau = 2$ respectively. The topology is completed by adding an initial straight

section to the serpentine as shown in Fig. 59(c). We define a vector $\vec{S} = (\tau_1, \tau_2, \dots, \tau_{|\mathbb{N}|})$ which contains the number of serpentine units $\tau_i \in \{1, \dots, \tau^{UB}\}$ for each subsystem, $i \in \mathbb{N}$. Here, τ_i can range from 1 to a user specified upper bound, τ^{UB} . Our experience indicates that a conservative value for τ^{UB} is 15, which corresponds to a serpentine composed of 61 individual channel sections where the number of sections = $4 \cdot \tau_i + 1$.

Relative Subsystem Positions. We encode the constraints to prevent subsystems from overlapping using a VLSI floor plan representation known as the Sequence Pair (SP) [108]. SP encodes the $\{left, right, above, below\}$ spatial relations between objects in the plane into a concise structure. This structure is readily searchable using standard heuristic methods such as Simulated Annealing (SA) [111].

A SP is composed of two \mathbb{N} -tuples $\Gamma = (\Gamma_+, \Gamma_-)$, where Γ_+ and Γ_- are ordered lists of the subsystem labels in \mathbb{N} . A SP maps to $|\mathbb{N}|(|\mathbb{N}| - 1)/2$ linear math programming constraints as shown in Eq. (74).

$$\begin{aligned} x_i + w_i &\leq x_j \leftarrow (\dots i \dots j \dots, \dots i \dots j \dots) \\ y_i + h_i &\leq y_j \leftarrow (\dots j \dots i \dots, \dots i \dots j \dots) \end{aligned} \tag{74}$$

Here, i and j are unique subsystem labels in \mathbb{N} . If i appears before j in both Γ_+ and Γ_- , then subsystem i is left of j . Likewise, if i appears after j in Γ_+ and before j in Γ_- , then subsystem i is below j . The SP representation has several attractive features. First, the solution space, although large, is finite i.e. $\Psi \propto (|\mathbb{N}|!)^2$. Second, the neighborhood of a particular arrangement A is readily constructable through simple perturbations of Γ . Furthermore, every evaluation of Γ results in a feasible planar arrangement, which is not true for all non-slicing floorplan representations (e.g. Corner Block List presented by Hong et al. [112]). Finally, SP is a general floorplan representation, which means that optimal solutions are not excluded due to assumptions about floorplan structure.

Well Placement. Confining the fluid I/O wells to the edges of the chip facilitates fluid transport on and off the chip. We have developed a graph-based approach that places a well on the chip edge closest to its associated port.

For a given arrangement A of subsystems, we construct an undirected, edge-weighted planar

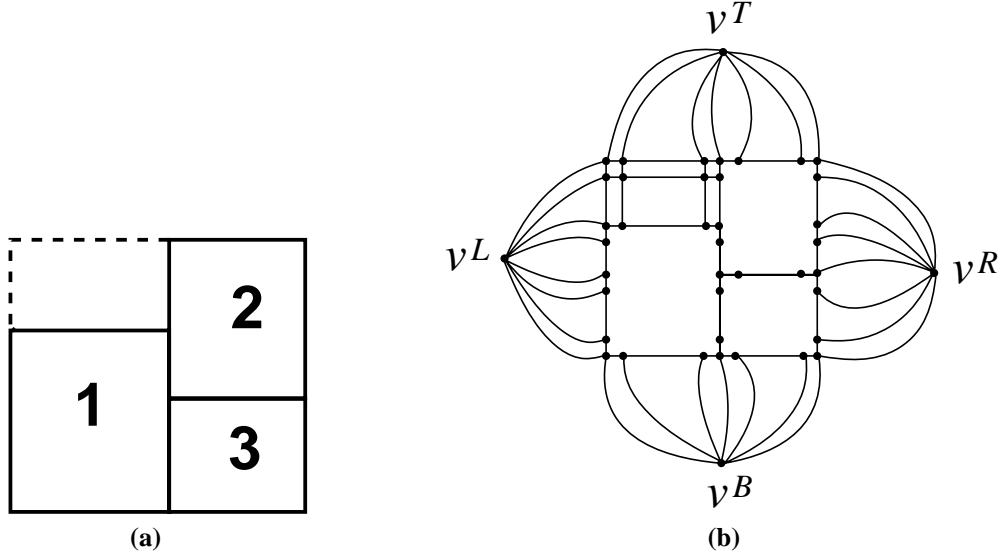


FIGURE 60. (a) Compacted arrangement of subsystems. (b) Corresponding well placement graph.

graph $G_A = \langle V, E \rangle$ by noting that a compacted arrangement of subsystems is essentially a planar graph (Fig. 60). The graph we create is similar to a *placement graph* used in VLSI circuit design [113].

We construct the graph by first generating a set of preliminary nodes by taking the Cartesian product of the bounding box dimensions, subsystem locations and port locations as shown in Eq. (75).

$$nodes = \{0, W, x_i, S_i^x, Sw_i^x, B_i^x, Bw_i^x; \forall i \in \mathbb{N}\} \times \{0, H, y_i, S_i^y, Sw_i^y, B_i^y, Bw_i^y; \forall i \in \mathbb{N}\} \quad (75)$$

A set of preliminary edges are formed by connecting each node with its nearest neighbor in the x and y directions. The graph's vertex set V and edge set E are generated by removing all the nodes and associated edges that reside within subsystem boundaries. Finally, four supernodes, v^L , v^T , v^R , and v^B are added to the vertex set. The supernodes are connected to the left-most, top-most, right-most, and bottom-most nodes of G_A as shown in Fig. 60(b).

The weights assigned to edges within the interior of G_A are simply the distances in the x and y directions between connected adjacent nodes. The edges connecting supernodes are weighted based on the maximum bounding box dimension. This prevents short-circuit paths through the graph during the well placement procedure. Each well is placed on one of the four sides of A as follows:

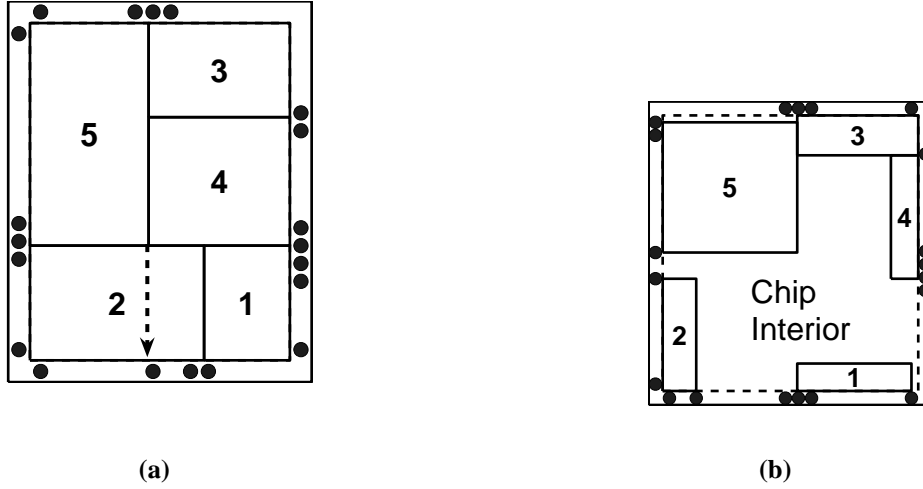


FIGURE 61. (a) Poor well placement based on rectilinear model (dotted arrow). (b) Compacted arrangement pulled apart by weighted-sum objective function.

1. Construct G_A and initialize sets: $left = right = top = bottom = \emptyset$.
2. Select a port vertex $p \in v_p$, where $v_p \subset V$ is the set of ports.
3. Determine the shortest paths from p to each supernode v^L, v^T, v^R, v^B . The shortest distance among these determines a connection from p to that edge. Store this distance in the set P .
4. Assign p to $left$ if p connects to v^L , to top if p connects to v^T , to $right$ if p connects to v^R or to $bottom$ if p connects to v^B .
5. Order wells in $left$ or $right$ based on the y -coordinates of their ports.
6. Order wells in top or $bottom$ based on the x -coordinates of their ports.
7. Obtain a total routing length estimate, ϕ , by summing the entries in P . (note that $|P| = |W| = 4 \cdot |\mathcal{S}|$)

In step 3, we apply a shortest path algorithm to efficiently search G_A . Steps 5 and 6 can be performed using an efficient sorting method.

Our solution methodology allows us to avoid two fundamental problems that would result from a more simplistic formulation. The first possible problem is illustrated in Fig. 61(a). Here wells are placed using a standard rectilinear distance metric. However, the shortest rectilinear distance between a port and a well, indicated by the dotted arrow, may pass through a subsystem and is therefore a poor estimate of the true distance between a port and the chip edge.

The distance estimates produced using G_A are far more accurate because they account for the fact that routes must go around subsystems.

A second problem can arise if a common VLSI floorplanning objective function of the

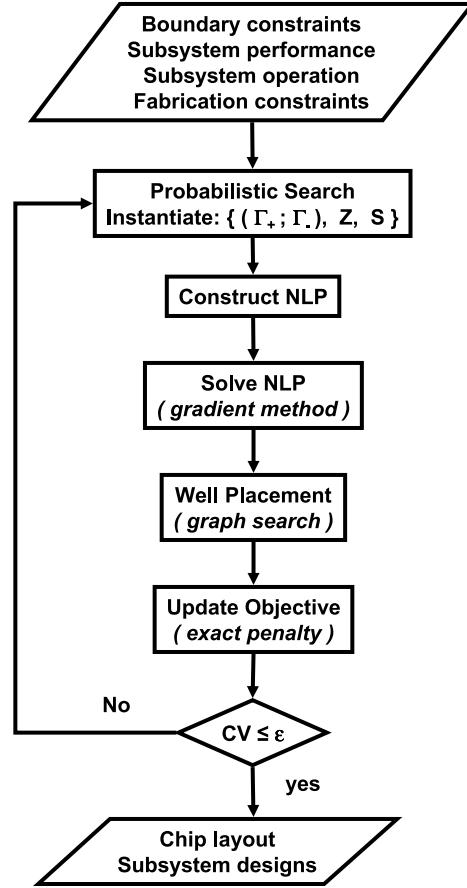


FIGURE 62. Flowchart for Floorplan Solution Method.

form: $\alpha_1 \cdot Area + \alpha_2 \cdot WL$ is used, where $Area$ is chip area, WL is wire length and α_1 and α_2 are weighting factors. This objective works well for typical circuit design problems because circuit elements are highly interconnected (i.e. wired together) within the chip interior and a wide variety of α_1 and α_2 values will produce satisfactory results. Since we connect single ports within the chip's interior to single wells on the chip's edges, over-weighting α_2 *pulls-apart* a compact arrangement and results in a large amount of unused space within the chip interior (Fig. 61(b)). Since we generally do not know good values for α_1 and α_2 *a priori*, we first compact the subsystems and lock their positions. Then we place the wells, thereby decoupling the continuous variables of the problem while retaining the global nature of the combinatoric variables, Γ , \vec{Z} and \vec{S} .

Floorplan Solution. A flowchart for the floorplanning algorithm is shown in Fig. 62. The main idea is to use a probabilistic search heuristic such as SA to deal with the combinatorial aspects of

the problem and an efficient gradient based method for the remaining continuous space problem. We have chosen this hybrid approach because SA has been shown to be an effective search method for difficult combinatorial problems, but is generally inferior to gradient-based methods on well posed continuous-space problems [114].

4.4.3. Routing of Auxiliary Microfluidic Channels

The goal of the routing stage is to take a compacted arrangement generated by our floorplan-
 ning algorithm and to determine the final placement or precise positions of subsystems and auxiliary channels that connect ports to wells. The auxiliary channels are kept as short and straight as possible to reduce fabrication costs and to allow for more convenient fluid loading and dispensing. In addition, short straight channels reduce dispersion effects [115] and concentration non-uniformity [26] as well as the operating voltage required to drive the chip.

Routing Problem Description. We define our routing problem as follows: Given a compact arrangement A of subsystems, find a routing solution which consists of the set of planar paths P through A such that each port is connected to a single well and that the total length and number of bends in P is minimized.

The routing of auxiliary channels is similar to wire routing in VLSI circuit design, however, there are several complicating features. Most importantly, LoC devices are generally fabricated in a single layer to reduce fabrication costs and complexity [115]. This means that all auxiliary channels must be routed in a planar fashion and can not be routed above or below a subsystem. Furthermore, the assumptions that channels can *feed through* a subsystem or that ports may move along a subsystem edge [113], do not apply in our formulation. Additionally, auxiliary channels occupy significant space on the chip, and therefore, the exact placement of routes is critical to the quality of the overall design.

The single-layer routing problem has been investigated in the VLSI literature [113]. However, much of this work is not directly relevant to our problem because it either pertains to global, multi-terminal, or detailed routing in bounded regions (e.g., channel routing). In global routing, the paths that wires must follow around obstacles on the chip are determined in a general way. A subsequent detailed routing stage places the wires precisely. Also, most VLSI routing algorithms construct multi-terminal routes, since single wires must often connect several terminals on the chip. Unlike channel-routing, which takes place in small bounded regions of the chip, we are interested in

finding routing solutions through the whole chip. Finally, many general VLSI routing algorithms employ a sequential *rip-up and re-route* approach to construct a routing solution. This approach lacks feasibility guarantees and can become artificially over-constrained when poor routing choices are made early in the procedure.

We are currently developing a routing procedure that is specifically designed to solve our two-terminal, single-layer, detailed routing problem. In addition, we directly address length and bend minimization. In our approach, we hope to exploit the network integrality property of the *minimum-cost network flow* (MCNF) formulation to find routing solutions in a simultaneous fashion using linear programming (LP). Large-scale MCNF problems can be solved using standard LP solvers or tailored algorithms [116]. Furthermore, LP can quickly provide rigorous feasibility guarantees. However, the bend-minimization constraints discussed in the following section break the MCNF structure, thus forcing us to use an integer programming (IP) approach. While many different IP formulations are possible to solve our routing problem, it is unclear which formulation will perform the best in all cases since there are non-trivial trade-offs between the number of constraints, the type of constraints and the number of variables.

We discuss our formulation, which is based loosely on the concept of MCNF, in the following section. We have successfully used this formulation to obtain good solutions for several small, compact test cases and are currently exploring algorithmic improvements that will allow us to deal with larger test cases.

4.4.4. Routing Solution Method

Fig. 63 shows a flowchart of our procedure for routing compacted arrangements. First, an initial arrangement is legalized so that it can be embedded in G_{grid} . Next we construct G_{net} from G_{grid} and use it to generate the associated integer program (IP). The IP is solved using a standard commercial solver [117]. If the IP is feasible, then the IP solver searches for an optimal routing. If an optimal routing or a routing that meets a user specified tolerance is found, then the algorithm terminates successfully. If the IP is infeasible, the design is expanded to open up new routing paths and is searched again. If the expanded design exceeds the available chip area, the floor plan algorithm must be queried for a new arrangement that may have better routing characteristics. Two of the key steps of our routing procedure are discussed in the following sections.

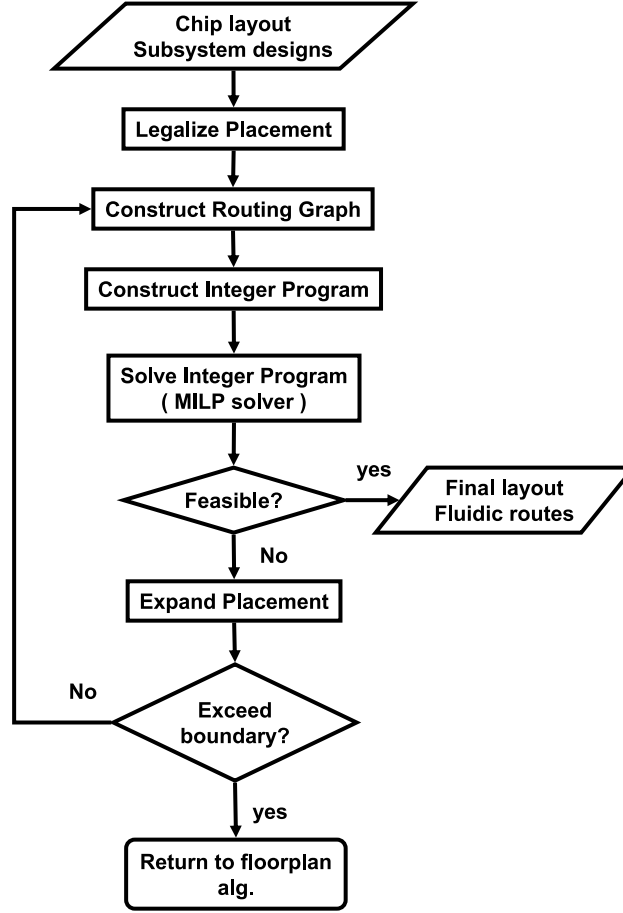


FIGURE 63. Flowchart for the routing algorithm.

Floorplan Legalization. Before an arrangement can be *placed* or embedded into G_{grid} , it must properly conform to the minimum allowable feature spacing requirement. If a subsystem's dimensions are not integer multiples of $PAD + \omega$, it cannot be properly embedded into the routing grid. Our floorplan algorithm does not guarantee that an arrangement will be embeddable. To account for this, we expand each subsystem in the x and y directions to the nearest integer multiple of $PAD + \omega$. For example, w_i is updated as follows: $w_i = \lceil w_i / (PAD + \Omega) \rceil \cdot (PAD + \Omega)$. The subsystem height h_i is treated in a similar fashion. The current Γ value of the floorplan is used to maintain the relative position of each subsystem. This procedure has a negligible effect on the performance and operation of each subsystem since the size-perturbation represents a relatively tiny fraction of the overall subsystem's size (typically $< 1.0\%$).

The wells on the chip's edges are updated in a similar fashion so that they too conform to valid grid points. As with other features on the chip, well edges are constrained to be no less than

PAD apart. After legalization, port, well and subsystem locations conform to vertices in G_{grid} . Therefore, when a routing solution is found, it represents the exact location of auxiliary channels on the chip.

Floorplan Expansion. For a given G_{net} , the solution to the IP formulation will either yield a feasible routing solution, or it will declare the current routing problem infeasible. If the IP is infeasible, then we are guaranteed that no routing solution exists for the current instance of G_{net} .

We have developed a procedure that iteratively augments G_{grid} by expanding the arrangement to increase the number of possible routing paths in the grid. Each expansion adds one additional routing path between each subsystem.

An arrangement is expanded by first conceptually expanding each subsystem's width and height by $\gamma(PAD + \omega)$. We define the arrangement expansion number γ as the number of times the arrangement is expanded. The arrangement's Γ is used to maintain the relative positions of each subsystem. Finally, each subsystem is shifted right and up from the origin by $\gamma(PAD + \omega)$. After each expansion, the arrangement is re-placed or re-embedded and a new G_{net} is generated. Expansions continue until either a feasible routing solution is obtained or until the placed design has become too large. Typically, arrangements require fewer than 5 expansions to achieve a feasible routing solution.

4.4.5. Multiplex Synthesis Results

Here we discuss some of the key features and algorithmic behavior of the floorplanning and routing algorithms. The results are encouraging with respect to both computation time and solution quality, since our implementations have not yet been optimized. However, these results also indicate important areas for improvement and future study.

The result of our floorplanning algorithm for our 5 subsystem example is shown in Fig. 64(a). This result was obtained in approximately 5 minutes of CPU time [118]. At this point, the design is compact and all constraints are completely satisfied.

The result of the routing algorithm for our 5 subsystem example is shown in Fig. 64(b). This design was routed in approximately 3 minutes of CPU time [118]. At this point, our design could be fabricated.

Floorplan Results. Fig. 65(a) shows a placed and compacted design including well positions for a

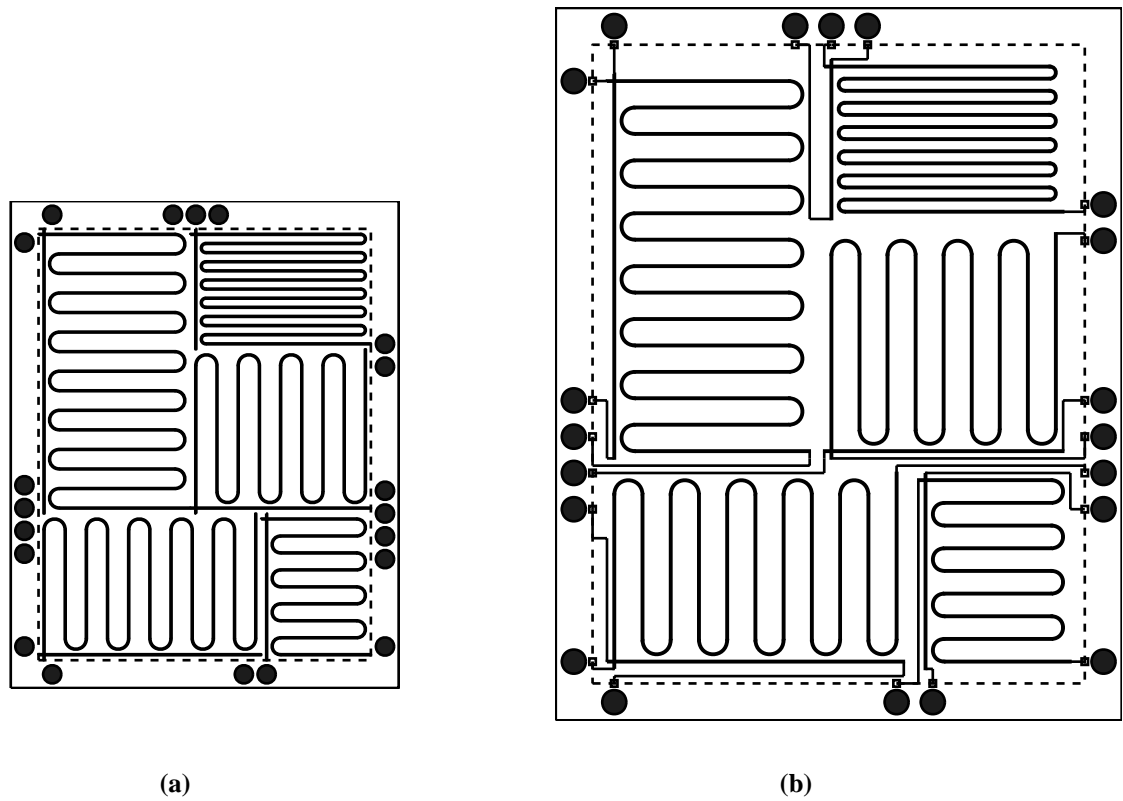


FIGURE 64. Five-subsystem example: (a) Compacted arrangement (b) Final design (routed).

multiplexed chip containing 10 subsystems. The completed design shown in Fig. 65(b), was generated in 2 hours on a standard PC [118] with equal time allocated to the floorplanning and routing

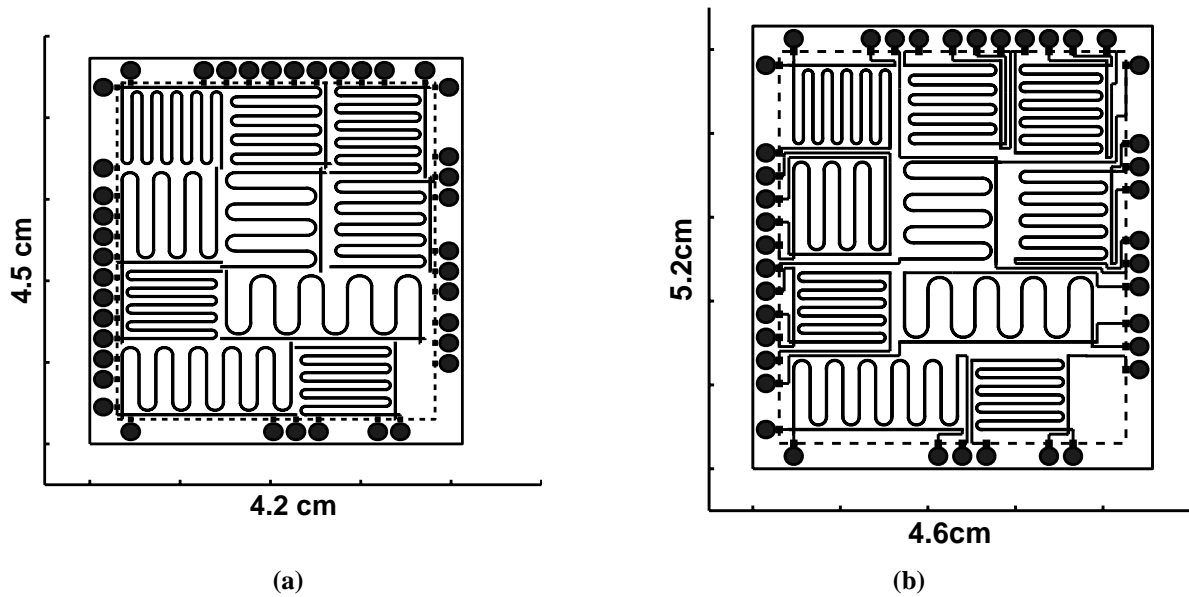


FIGURE 65. Ten-subsystem example: (a) Compacted arrangement (b) Final routed design.

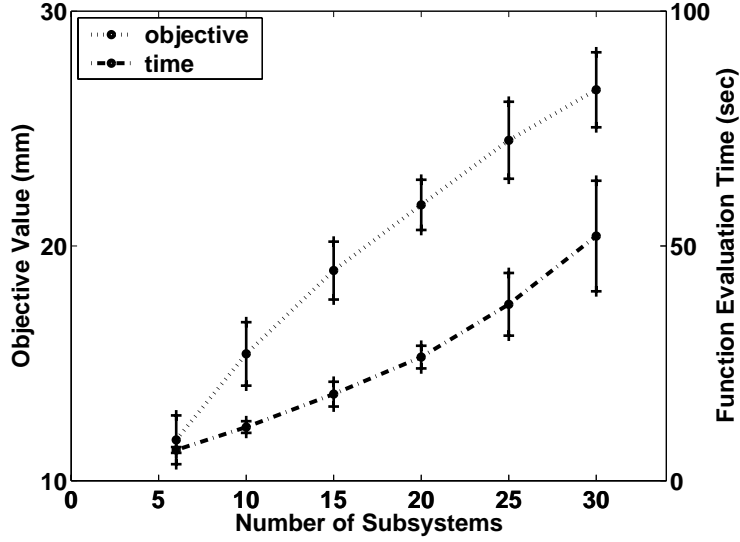


FIGURE 66. Scaling of NLP compaction subproblem in the floorplanning algorithm per size of design instance. Tests were performed on a standard PC (Intel Pentium 3 CPU: 1GHz CPU, 1Gb RAM).

stages. While we are pleased with these results, significantly more time was required to generate the 10 subsystem design than was required for the 5 subsystem design. The compaction NLP presents the most significant computational bottleneck during the floorplanning stage.

It appears that 80 % - 90 % of the computation time used by our floorplanning algorithm is spent solving the compaction NLP. Fig. 66 shows the average change in objective value (*dotted line*) and the average function evaluation time (*dash & dotted line*) needed to solve the compaction NLP versus the number of subsystems in a given design. We created our test cases by first choosing a common set of typical physical property values, operating conditions and performance specifications for each subsystem. Each data point represents the average of 20 randomly generated instances. One standard deviation about the mean is also shown.

As expected, the standard deviations of both time and objective value increase as the number subsystems increase. This is because the solution space grows in proportion to $(|N|!)^2 \cdot 8^{|N|}$.

It appears from the tests we have conducted that our algorithm scales reasonably. This can be attributed to the success of our initialization procedure, which helps the NLP solver converge to locally optimal solutions in reasonable time. While the compaction NLP appears to scale acceptably, the SA must evaluate the NLP more often as designs become larger-scale to achieve satisfactory results. We have produced multiplexed serpentine designs containing up to 30 subsystems in under one day. As far as we know, these designs are significantly larger than any multiplexed ser-

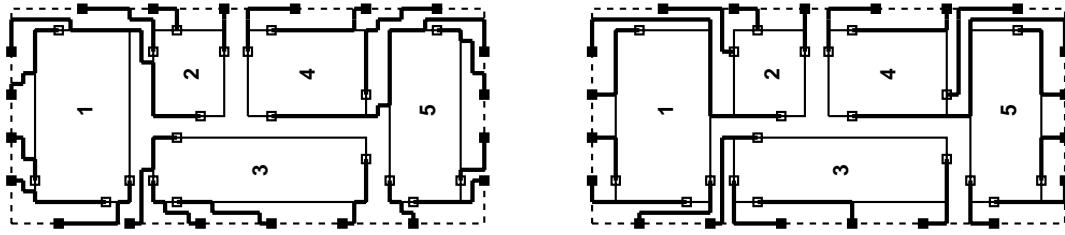


FIGURE 67. Example routing: (a) Minimum length only. (b) Minimum length and number of bends.

pentine designs presented in the literature. However, it is apparent that as problem size continues to increase, the computation time required by our algorithm will become prohibitive. We are currently investigating methods that will allow us to solve larger scale problems within a reasonable amount of time.

Microchannel Routing Results. Fig. 67(a) shows a minimum length routing for a simple 5 subsystem floorplan. The thick lines are the auxiliary channels connecting ports to wells. This floorplan has been expanded two times (i.e. $\gamma = 2$). This means that there are at least three possible routing paths between each subsystem (each subsystem edge and one path in between).

Notice that since there are many degenerate equal-length paths between ports and wells, the routing solution contains many bends. Fig. 67(b) shows a routing solution for the same example when both the channel length and the number of bends have been minimized. Typically, both the total channel length and the port-well assignment remain unaltered between the minimum length routing and the minimum length and bend routing (although this behavior is not guaranteed). This is because bend minimization represents only a small fraction of the overall objective value. In general, it does not cause re-routing to occur.

While bend minimization helps to reduce the number of non-unique solutions, it increases the optimality gap and makes the problem more computationally difficult. In our experiments, a standard IP solver [117] can typically prove a proposed routing grid infeasible in less than 30 seconds. Therefore the decision to expand the grid is made efficiently. When bend minimization is added to our problem, obtaining a proven optimal solution often becomes computationally prohibitive. Furthermore, no routing computation-time correlation can be drawn based on the number of subsystems in an arrangement since the routing problem scales with the number of grid edges and not the number of subsystems. Currently, we cope with these issues by allowing routing solutions to

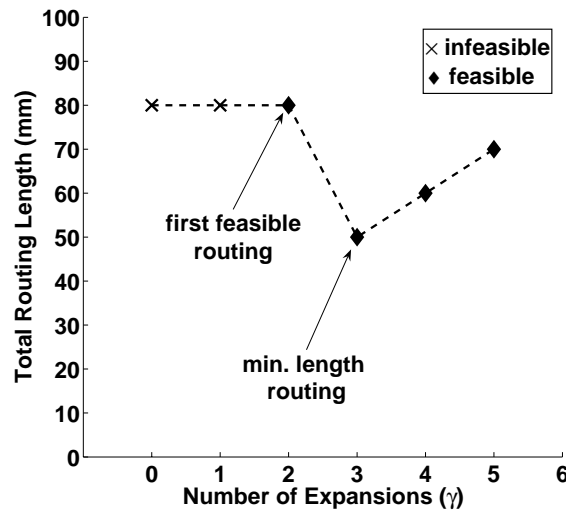


FIGURE 68. Total routing length vs. number expansions.

have a small relative optimality gap. Despite these complications, we have obtained reasonable quality solutions in 1 hour of CPU time [118] for 10 subsystem designs.

Fig. 68 shows the typical operation of our routing algorithm for the 5 subsystem example. It illustrates the trade-off between minimum channel length and minimum design area. Fig. 68 depicts how iteratively expanding a design influences the total auxiliary channel length. For expansion numbers of $\gamma = 0$ and $\gamma = 1$, no feasible routing can be found. A feasible routing is obtained at $\gamma = 2$.

Normally, we would terminate the algorithm after optimizing the first feasible routing. However, if the design is expanded a third, fourth and fifth time, we notice that we achieve what appears to be a globally minimal length routing at $\gamma = 3$. This occurs because for low γ 's, the routing grid is highly constrained. Therefore, the first feasible routing solution will often contain long channels. As γ increases, more possible routing paths are created and shorter routes become available. Eventually, continuing to increase γ will result in longer channels simply because the design is now larger and the channels must traverse a longer distance to connect ports to wells.

The minimum length solution at $\gamma = 3$ is also significant because this solution most closely approaches the solution obtained by the well placement algorithm described earlier. Recall that in the well placement algorithm, route continuity is enforced, but route planarity is not enforced. The well placement algorithm is in fact a lower bound relaxation of our IP routing formulation. We are currently investigating how to utilize this information to help us more quickly discover optimal or near optimal routing solutions.

4.4.6. Summary

We have presented a design automation methodology that is capable of generating full-custom multiplexed LoC's for CE applications in only minutes to hours. We have adapted and combined VLSI circuit design techniques with optimal design methods to perform LoC design and layout. We are currently investigating methods for parallelizing our floorplanning algorithm to handle larger problem instances. We are also investigating ways to reduce the number of edges in routing grids while still maintaining a high level of connectivity. We believe that our experience with multiplexed design provides us with a tool set that can be extended to handle multifunction chips that contain many complex chemical operations.

5. Conclusions

The primary conclusion of this work is that it is possible to develop biofluidic microsystem models for optimization-based design synthesis. The separation, mixing and reaction models developed in this project are capable to meet the needs of iterative simulations without any additional augmenting information from continuum simulation. At the start of this project the state of the art was parameterized lumped element models for simple geometries, and reduced order models for more complicated geometries. This project has extended the boundary for which parameterized lumped element models are available as well as extended the applicable range of design dimensions. For example, the models and the accompanying simulation capability were able to predict the performance of a highly convective dispersion in a hybrid electrophoresis microchip.

Component and system optimization using these models were demonstrated. They invariably improved the quality of the design compared to the existing design published in the open literature.

6. Recommendations

Further development of application-specific biofluidic microsystem design should proceed. the suite of modeling, simulation and optimization tools developed in this project are now a basic design framework for design improvement. An NHGRI supported project on desinging DNA sequencing microchips will be the first area where this project's investigators will look at. Other DoD targets may be potentially identified by one of the graduated students from the project who is now working with Northrop Grumman or by another of the graduated students working at CFDRC.

7. References

- [1] D. Jed Harrison, et. al., "From Micro-Motors to Micro-Fluidics: The Blossoming of Micromachining Technologies in Chemistry, Biochemistry and Biology," *Transducers '99*, Sendai, Japan.
- [2] P. Griss and G. Stemme, "Side-opened out-of-plane microneedles for microfluidic transdermal liquid transfer," *IEEE Journal of Microelectromechanical Systems*, vol. 12, no. 3, pp. 296-301, Jun 2003. Image obtained from <http://www.s3.kth.se/mst/research/index.shtml>.
- [3] C. A. Emrich, H. J. Tian, I. L. Medintz, and R. A. Mathies, "Microfabricated 384-lane capillary array electrophoresis bioanalyzer for ultrahigh-throughput genetic analysis," *Anal. Chem.*, v. 74, pp. 5076-5083, Oct. 2002. Image from <http://www.cchem.berkeley.edu/ramgrp/alpha/>.
- [4] Affymetrix GeneChip Essentials, http://www.affymetrix.com/corporate/media/genechip_essentials/gene_expression/An_Actual_Gene_Expression_Image.affx
- [5] A. B. Kahng, I. I. Mandoiu, S. Reda, X. Xu, and A. Zelikovsky, "Evaluation of placement techniques for DNA probe array layout," *Proc. IEEE/ACM ICCAD*, Nov. 2003, pp. 262-269.
- [6] A. B. Kahng, I. I. Mandoiu, S. Reda, X. Xu, and A. Zelikovsky, "Computer-Aided Optimization of DNA Array Design and Manufacturing," *IEEE Transactions on Computer-Aided Design*, Feb. 2006, pp. 305-320.
- [7] Agilent Microarrays, <http://www.chem.agilent.com>.
- [8] Nanogen, <http://www.nanogen.com/technology/index.htm>.
- [9] R. C. Anderson, G. J. Bogdan, A. Puski, X. Su, "Genetic analysis systems: improvements and methods," *Tech. Dig. of Solid-State Sensor and Actuator Workshop*, Hilton Head Isl, SC, Jun 7-11 1998, pp. 7-10.
- [10] A. Manz, N. Graber, and H. M. Widmer, "Miniaturized Total Chemical-Analysis Systems - a Novel Concept for Chemical Sensing," *Sensors and Actuators B-Chemical*, vol. 1, pp. 244-248, Jan. 1990.
- [11] A. A. Darhuber and S. M. Troian, "Principles of microfluidic actuation by modulation of surface stresses," *Annual Review of Fluid Mechanics*, vol. 37, pp. 425-455, Jan. 2005.
- [12] M. G. Pollack, A. D. Shenderov, and R. B. Fair, "Electrowetting-based actuation of droplets for integrated microfluidics," *Lab on a Chip*, vol. 2, pp. 96-101, 2002.
- [13] S. K. Cho, H. J. Moon, and C. J. Kim, "Creating, transporting, cutting, and merging liquid droplets by electrowetting-based actuation for digital microfluidic circuits," *Journal of Microelectromechanical Systems*, vol. 12, pp. 70-80, Feb. 2003.
- [14] S.-K. Fan, C. Hashi, C.-J. Kim, "Manipulation of multiple droplets on NxM grid by cross-reference EWOD driving scheme and pressure-contact packaging," *IEEE MEMS '03*, pp. 694-697, Kyoto, Japan.
- [15] V. Srinivasan, V. K. Pamula, and R. B. Fair, "An integrated digital microfluidic lab-on-a-chip for clinical diagnostics on human physiological fluids," *Lab on a Chip*, pp. 310-315, 2004.
- [16] H. Moon, A. R. Wheeler, R. L. Garrell, J. A. Igo, and C.-J. Kim, "On-Chip Sample Preparation by Electrowetting-on-Dielectric Digital Microfluidics for Matrix Assisted Laser Desorption/Ionization Mass Spectrometry," *IEEE MEMS '05*, Miami, FL, pp. 859-862.
- [17] J. Vykoukal, J. Schwartz, F. Becker, and P. R. C. Gascoyne, "A programmable dielectrophoretic fluid processor for droplet-based chemistry," in *Proc. Micro Total Analysis Systems (μ TAS 2001)*, pp. 72-74.
- [18] M. Washizu, "Manipulation of biological objects in micromachined structures," *IEEE MEMS '92*, pp. 196-201.
- [19] P. R. C. Gascoyne, and J. V. Vykoukal, "Dielectrophoresis-based sample handling in general-

- purpose programmable diagnostic instruments,” *Proceedings of the IEEE*, vol. 92, no. 1, pp. 22-42, Jan 2004.
- [20]C. T. Culbertson, S. C. Jacobson, and J. M. Ramsey, “Microchip devices for high-efficiency separations,” *Analytical Chemistry*, vol. 72, pp. 5814-5819, Dec 2000.
- [21]P. Watts and S. J. Haswell, “The application of micro reactors for organic synthesis,” *Chem. Soc. Rev.*, 2005, 34, pp. 235–246.
- [22]N. H. Chiem and D. J. Harrison, “Microchip systems for immunoassay: an integrated immunoreactor with electrophoretic separation for serum theophylline determination,” *Clinical Chemistry*, vol. 44, pp. 591-598, Mar. 1998.
- [23]P. M. S. John, T. Woudenberg, C. Connell, M. Deshpande, J. R. Gilbert, M. Garguilo, P. Paul, J. Molho, A. E. Herr, T. W. Kenny, and M. G. Mungal, “Metrology and Simulation of Chemical Transport in Microchannels,” *Tech. Dig. of Solid-State Sensor and Actuator Workshop*, Hilton Head Isl, SC, pp. 106-111, June 7-11, 1998.
- [24]P. Stout, H.Q. Yang, P. Dionne, A. Leonard, Z. Tan, A. Przekwas, A. Krishnan, “CFD-ACE+MEMS: a CAD system for simulation and modeling of MEMS,” *Proc. Design, Test, Integration and Packaging of MEMS/MOEMS (DTIP '99)*, Paris, France, pp 328-339.
- [25]J. Zeng and T. Korsmeyer, “Principles of droplet electrohydrodynamics for lab-on-a-chip,” *Lab Chip*, 2004, (4), 265-277.
- [26]J. I. Molho, A. E. Herr, B. P. Mosier, J. G. Santiago, T. W. Kenny, R. A. Brennen, G. B. Gordon, and B. Mohammadi, “Optimization of turn geometries for microchip electrophoresis,” *Analytical Chemistry*, vol. 73, pp. 1350-1360, Mar 15 2001.
- [27]J. White, “CAD Challenges in BioMEMS Design,” *Proc. DAC 2004, Proc. DAC 2004*, San Diego, CA, Jun 7-11, 2004, pp. 629-632.
- [28]X. Wang, M. Judy, J. White, “Validating fast simulation of air damping in micromachined devices,” *IEEE MEMS '02*, pp. 210-213.
- [29]Y. Wang, R. Magargle, Q. Lin, J. Hoburg and T. Mukherjee, “System-Oriented Modeling and Simulation of Biofluidic Lab-on-a-Chip,” *13th Intl. Conf. on Solid-State Sensors, Actuators and Microsystems (TRANSDUCERS '05)*, pp. 1280-1283, June 5-9, 2005, Seoul, Korea.
- [30]M. Rewienski, and J. White, “A Trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices,” *IEEE Transactions of Computer-Aided Design*, 22(2):155-170, 2003.
- [31]R. Qiao and N.R.Aluru, “A compact model for electroosmotic flows in microfluidic devices,” *Journal of Micromechanics and Microengineering*, v 12, n 5, Sept. 2002, p 625-35.
- [32]A.N. Chatterjee and N.R.Aluru, “Combined circuit/device modeling and simulation of integrated microfluidic systems,” *Journal of Microelectromechanical Systems*, v 14, n 1, Feb. 2005, p 81-95.
- [33]T. Korsmeyer, J. Zeng and K. Greiner, “Design Tools for BioMEMS,” *Proc. DAC 2004*, San Diego, CA, Jun 7-11, 2004, pp. 622-627.
- [34]B. Baidya, T. Mukherjee and J. F. Hoburg, “Dispersion Modeling in Microfluidic Channels for System-level Optimization,” in *2002 Intl. Conf. on Modeling and Simulation of Microsystems (MSM '02)*, San Juan, Puerto Rico, April 22-25 2002, pp. 182-185.
- [35]Y. Wang, Q. Lin and T. Mukherjee, “Applications of Behavioral Modeling and Simulation on Lab-on-a-Chip: Micro-Mixer and Separation System,” *Int. Workshop on Behavioral Modeling and Simulation (BMAS '04)*, pp. 8-13, October 21-22, 2004, San Jose, California.
- [36]R. F. Probstein, *Physicochemical Hydrodynamics : An Introduction*, 2nd ed. New York: John Wiley & Sons, 2003.

- [37]Y. Wang, Q. Lin, and T. Mukherjee, "System-oriented dispersion models of general-shaped electrophoresis microchannels," *RSC Lab on a Chip*, vol. 4, pp. 453-463, Oct. 2004.
- [38]H. Spiess and P. Wriggers, "Reduction Methods for FE Analysis in Nonlinear Structural Dynamics," *PAMM*, vol. 5, pp. 135-136, 2005.
- [39]G. Mazzitelli and M. Mattioli, "Ionization Balance for Optically Thin Plasmas: Rate Coefficients for Cu, Zn, Ga, and Ge Ions," *Atomic Data and Nuclear Data Tables*, vol. 82, pp. 313-356, 2002.
- [40]R. Magargle, J. F. Hoburg and T. Mukherjee, "Microfluidic Injector Models Based on Neural Networks," in *IEEE Transactions on CAD*, Special Issue on Design Automation Methods and Tools for Microfluidics-based Biochips, February 2006, pp. 378-385.
- [41]O. Mikulchenko and K. Mayaram, "Neural Network Design for Behavioral Model Generation with Shape Preserving Properties," *Behavioral Modeling and Simulation Conference*, 2000.
- [42]H. Zaabab, Q. J. Zhang, and M. Nakhla, "Analysis and Optimization of Microwave Circuits and Devices Using Neural Network Models," *IEEE International Microwave Symposium Digest*, pp. 393-396, 1994.
- [43]MATLAB, The MathWorks, Inc. <http://www.mathworks.com>, 2006.
- [44]C. Bishop, *Neural networks for pattern recognition*. New York: Oxford University Press, 1995.
- [45]A. R. Barron, "Universal Approximation Bounds for Superpositions of a Sigmoidal Function," *IEEE Transactions on Information Theory*, vol. 39, pp. 930-945, 1993.
- [46]T. Hastie, R. Tibshirani, and J. H. Friedman, *The elements of statistical learning : data mining, inference, and prediction*. New York: Springer, 2001.
- [47]S. Geman, E. Bienenstock, and R. Doursat, "Neural Networks and the Bias/Variance Dilemma," *Neural Computation*, vol. 4, pp. 1-58, 1992.
- [48]I. M. Sobol, "On the distribution of points in a cube and the approximate evaluation of integrals," *USSR Computational Mathematics and Mathematical Physics*, vol. 7, pp. 86-112, 1967.
- [49]I. M. Sobol, "Uniformly distributed sequences with additional uniformity properties," *USSR Computational Mathematics and Mathematical Physics*, vol. 16, pp. 236-242, 1976.
- [50]H. Faure, "Discrepance de suites associees a un systeme de numeration (en dimension s)." *Acta. Arithmetica* XLI, pp. 337-351, 1982.
- [51]J. H. Halton, "On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals.," *Numerical Mathematics*, vol. 2, 1960.
- [52]J. Hammersley, "Monte Carlo methods for solving multivariable problems," *Proceedings of the New York Academy of Science*, vol. 86, pp. 844-874, 1960.
- [53]H. Niederreiter, "Point sets and sequences with small discrepancy," *Monatshefte fur Mathematik*, vol. 104, pp. 273-337, 1987.
- [54]P. Bratley and B. Fox, "Algorithm 659: Implementing Sobol's quasirandom sequence generator," *ACM Transactions on Mathematical Software*, vol. 14, pp. 88-100, 1988.
- [55]SobolSeq, British-Russian Offshore Development Agency (BRODA), <http://www.broda.co.uk/software.html>, 2006.
- [56]P. R. Gill, W. Murray, and M. H. Wright, *Practical Optimization*. London: Academic Press, 1981.
- [57]Y. Wang, Q. Lin and T. Mukherjee, "Composable Behavioral Models and Schematic-Based Simulation of Electrokinetic Lab-on-a-Chips," in *IEEE Transactions on CAD*, Special Issue on Design Automation Methods and Tools for Microfluidics-based Biochips, February 2006, pp. 258-273.
- [58]P. Lob, K. S. Drese, V. Hessel, S. Hardt, C. Hofmann, H. Lowe, R. Schenk, F. Schonfeld, and B.

- Werner, "Steering of liquid mixing speed in interdigital micro mixers - from very fast to deliberately slow mixing," *Chemical Engineering & Technology*, vol. 27, pp. 340-345, 2004.
- [59] Y. Wang, Q. Lin, and T. Mukherjee, "A Model for Laminar Diffusion-Based Complex Electrokinetic Passive Micromixers," *RSC Lab on Chip*, vol. 5, no. 8, pp. 877-887, Aug. 2005.
- [60] S. C. Jacobson, T. E. McKnight, and J. M. Ramsey, "Microfluidic devices for electrokinetically driven parallel and serial mixing," *Analytical Chemistry*, vol. 71, pp. 4455-4459, 1999.
- [61] Y. Wang, Q. Lin and T. Mukherjee, "A Model for Joule Heating-Induced Dispersion in Microchip Electrophoresis," *Lab on a Chip*, 4: 625-631, 2004
- [62] S. C. Jacobson, R. Hergenroder, L. B. Koutny, R. J. Warmack, and J. M. Ramsey, "Effects of Injection Schemes and Column Geometry on the Performance of Microchip Electrophoresis Devices," *Analytical Chemistry*, vol. 66, pp. 1107-1113, 1994.
- [63] Cadence Design Systems, <http://www.cadence.com>.
- [64] S. C. Jacobson and J. M. Ramsey, "Electrokinetic focusing in microfabricated channel structures," *Analytical Chemistry*, vol. 69, pp. 3212-3217, 1997.
- [65] J. B. Knight, A. Vishwanath, J. P. Brody, and R. H. Austin, "Hydrodynamic focusing on a silicon chip: Mixing nanoliters in microseconds," *Physical Review Letters*, vol. 80, pp. 3863-3866, 1998.
- [66] C. T. Culbertson, S. C. Jacobson, and J. M. Ramsey, "Dispersion sources for compact geometries on microchips," *Analytical Chemistry*, vol. 70, pp. 3781-3789, 1998.
- [67] S. K. Griffiths and R. H. Nilson, "Design and analysis of folded channels for chip-based separations," *Analytical Chemistry*, vol. 74, pp. 2960-2967, 2002.
- [68] B. M. Paegel, L. D. Hutt, P. C. Simpson, and R. A. Mathies, "Turn geometry for minimizing band broadening in microfabricated capillary electrophoresis channels," *Analytical Chemistry*, vol. 72, pp. 3030-3037, 2000.
- [69] D. J. Harrison, A. Manz, Z. H. Fan, H. Ludi, and H. M. Widmer, "Capillary Electrophoresis and Sample Injection Systems Integrated on a Planar Glass Chip," *Analytical Chemistry*, vol. 64, pp. 1926-1932, 1992.
- [70] A. Manz, D. J. Harrison, E. M. J. Verpoorte, J. C. Fettingner, A. Paulus, H. Ludi, and H. M. Widmer, "Planar Chips Technology for Miniaturization and Integration of Separation Techniques into Monitoring Systems - Capillary Electrophoresis on a Chip," *Journal of Chromatography*, vol. 593, pp. 253-258, 1992.
- [71] Z. Liang, N. Chiem, G. Ocirk, T. Tang, K. Fluri, and D. J. Harrison, "Microfabrication of a Planar Absorbance and Fluorescence Cell for Integrated Capillary Electrophoresis Devices," *Analytical Chemistry*, vol. 68, pp. 1040-1046, 1996.
- [72] D. J. Harrison, K. Fluri, K. Seiler, Z. Fan, C. S. Effenhauser, and A. Manz, "Micromachining a Miniaturized Capillary Electrophoresis-Based Chemical Analysis System on a Chip," *Science*, vol. 261, pp. 895-897, 1993.
- [73] C. S. Effenhauser, A. Manz, and H. M. Widmer, "Glass Chips for High-Speed Capillary Electrophoresis Separations with Submicrometer Plate Heights," *Analytical chemistry*, vol. 65, pp. 2637, 1993.
- [74] C. S. Effenhauser, A. Paulus, A. Manz, and H. M. Widmer, "High-Speed Separation of Antisense Oligonucleotides on a Micromachined Capillary Electrophoresis Device," *Analytical Chemistry*, vol. 66, pp. 2949-2953, 1994.
- [75] F. v. Heeren, E. Verpoorte, A. Manz, and W. Thormann, "Characterization of Electrophoretic Sample Injection and Separation in a Gell-Filled Cyclic Planar Microstructure," *Journal of Microcolumn Separations*, vol. 8, pp. 373-381, 1996.

- [76]S. V. Ermakov, S. C. Jacobson, and J. M. Ramsey, "Computer simulations of electrokinetic injection techniques in microfluidic devices," *Analytical Chemistry*, vol. 72, pp. 3512-3517, 2000.
- [77]Y. Liu, R. S. Foote, S. C. Jacobson, R. S. Ramsey, and J. M. Ramsey, "Electrophoretic Separation of Proteins on a Microchip with Noncovalent, Postcolumn Labeling," *Analytical Chemistry*, vol. 72, pp. 4608-4613, 2000.
- [78]J. P. Kutter, S. C. Jacobson, N. Matsubara, and J. M. Ramsey, "Solvent-Programmed Microchip Open-Channel Electrochromatography," *Analytical Chemistry*, vol. 70, pp. 3291-3297, 1998.
- [79]A. G. Hadd, S. C. Jacobson, and J. M. Ramsey, "Microfluidic Assays of Acetylcholinesterase Inhibitors," *Analytical Chemistry*, vol. 71, pp. 5206-5212, 1999.
- [80]R.-J. Yang, L.-M. Fu, and G.-B. Lee, "Variable-Volume-Injection Methods Using Electrokinetic Focusing on Microfluidic Chips," *Journal of Separation Science*, vol. 25, pp. 996-1010, 2002.
- [81]L.-M. Fu, R.-J. Yang, G.-B. Lee, and Y.-J. Pan, "Multiple Injection Techniques for Microfluidic Sample Handling," *Electrophoresis*, vol. 24, pp. 3026-3032, 2003.
- [82]C.-H. Tsai, R.-J. Yang, C.-H. Tai, and L.-M. Fu, "Numerical Simulation of Electrokinetic Injection Techniques in Capillary Electrophoresis Microchips," *Electrophoresis*, vol. 26, pp. 674-686, 2005.
- [83]C.-X. Zhang and A. Manz, "Narrow Sample Channel Injector for Capillary Electrophoresis on Microchips," *Analytical Chemistry*, vol. 73, pp. 2656-2662, 2001.
- [84]S. Rudolph, "On Topology, Size and Generalization of Non-Linear Feed-Forward Neural Networks," *Neurocomputing 1997*, Vol. 16, pp. 1-22
- [85]S. J. Haswell and P. Watts, "Green chemistry: Synthesis in microreactors," *Green Chem.*, 5:240-249, 2003.
- [86]W. Ehrfeld, V. Hessel, and H. Lehr, *Microreactors: New Technology for Modern Chemistry*. Wiley-VCH, 2000.
- [87]S. K. Ajmera, C. Delattre, M.A. Schmidt, and K.F. Jensen, "Microfabricated differential reactor for heterogenous gas phase catalyst testing," *J. Catal.*, 209:401-412, 2002.
- [88]V. Skelton, G.M. Greenway, S.J. Haswell, P. Styring, D.O. Morgan, B. Warrington, and S.Y.F. Wong, "The generation of concentration gradients using electroosmotic flow in micro reactors allowing stereoselective chemical synthesis," *Analyst*, 126:11-13, 2002.
- [89]S. K. Ajmera, C. Delattre, M.A. Schmidt, and K.F. Jensen, "Microfabricated cross-flow chemical reactor for catalyst testing," *Sens.Actuators B*, 82:297-306, 2002.
- [90]P. D. I. Fletcher, S.J. Haswell, and V.N. Paunov, "Theoretical considerations of chemical reactions in micro-reactors operating under electroosmotic and electrophoretic control," *Analyst*, 124:1273-1282, 1999.
- [91]P. D. I. Fletcher, S.J. Haswell, and X. Zhang, "Electrical currents and liquid flow rates in micro-reactors," *Lab on a Chip*, 1:115-121, 2001.
- [92]COMSOL FEMLAB, <http://www.femlab.com>.
- [93]A. J. Pfeiffer, T. Mukherjee, and S. Hauan, "Synthesis of multiplexed biofluidic microchips." *Trans. on CAD*, 25:321-333, 2006.
- [94]N. A. Patankar and H.H. Hu, "Numerical simulation of electroosmotic flow," *Anal.Chem.*, 70:1870-1881, May 1998.
- [95]E. B. Cummings, S.K. Griffiths, R.H. Nilson, and P.H. Paul, "Conditions of similitude between the fluid velocity and electric field in electroosmotic flow," *Anal.Chem.*, 72:2526-2532, 2000.
- [96]J. M. MacInnes, X. Du, and R.W.K. Allen, "Dynamics of electroosmotic switching of reacting microfluidic flows," *Trans IChemE*, 2003.

- [97]W. E. Schiesser, *The Numerical Method of Lines: Integration of Partial Differential Equations*. Academic Press, Inc., 1991.
- [98]K. Gustafson. *Introduction to Partial Differential Equations and Hilbert Space Methods*. Dover, 1999.
- [99]R. B. Bird, E. N. Lightfoot, and W. E. Stewart, *Transport phenomena*, 2nd ed. New York: J. Wiley, 2002.
- [100]S. J. Haswell, "Development and operating characteristics of micro flow injection analysis systems based on electroosmotic flow - A review," *Analyst*, vol. 122, pp. R1- R10. 1997.
- [101]A. Pfeiffer, T. Mukherjee and S. Hauan, "Design and Optimization of Compact Microscale Electrophoretic Separation Systems", *Ind. Eng. Chem. Res.*, p. 3539, vol. 43(14), (2004).
- [102]A. J. Pfeiffer, "Lab-on-a-Chip Design Automation," Ph.D. Thesis, Chemical Engineering Dept., Carnegie Mellon University, 2006.
- [103]C. A. Floudas and P.M. Pardalos, editors. *Encyclopedia of Optimization*. Kluwer Academic Publishers, 2001.
- [104]Micronit Microfluidics. Micronit: microsystems foundry. <http://www.micronit.com>, June 2003.
- [105]A. J. Pfeiffer, T. Mukherjee and S. Hauan, "Synthesis of Multiplexed Biofluidic Microchips," in *IEEE Transactions on CAD*, Special Issue on Design Automation Methods and Tools for Microfluidics-based Biochips, February 2006, pp. 321-333.
- [106]T. Lengauer. *Combinatorial algorithms for integrated circuit layout*. Wiley-Teubner, 1990.
- [107]S. Lee and I.E. Grossmann, "New algorithms for nonlinear generalized disjunctive programming," *Comp.Chem.Engng.*, 24(9–10):2125–2141, 2000.
- [108]H. Murata, K. Fujiyoshi, S. Nakatake, and Y. Kajitani, "VLSI module placement based on rectangle-packing by the Sequence Pair," *Trans. on CAD*, 15(12):1518–1524, 1996.
- [109]H. Onodera, Y. Taniguchi, and K. Tamaru, "Branch-and-bound placement for building block layout," *In Proceedings of Design Automation Conference (DAC)*, pages 433-439, 1991.
- [110]S. Sutanthavibul, E. Shragowitz, and J.B. Rosen, "An analytical approach to floorplan design and optimization," *Trans. on CAD*, 10(6):761-769, 1991.
- [111]S. Kirkpatrick, C.D. Galett, and M.P Vecchi, "Optimization by simulated annealing," *Science*, 13 May 1983, 220, 4598:671-680, 1983.
- [112]X. Hong, G. Huang, Y. Cai, J. Gu, S. Dong, C-K. Cheng, and J. Gu, "Corner block list: an effective and efficient topological representation of non-slicing floorplan," *In Proceedings of International Conference on Computer Aided Design (ICCAD)*, pages 8-12, 2000.
- [113]M. Sarrafzadeh and C.K. Wong, *An Introduction to VLSI Physical Design*, McGraw-Hill Inc., 1996.
- [114]P. Y. Papalambros and D.J. Wilde, *Principles of Optimal Design: Modeling and Computation*, Cambridge University Press, 2000.
- [115]LioniX. LioniX: MEMS foundry services. <http://www.lionixbv.nl>, November 2004.
- [116]H. P. Williams, *Model Building in Mathematical Programming*, John Wiley and Sons, 4 edition, 1999.
- [117]ILOG Inc, ILOG CPLEX 9.0.: integer programming solver, <http://www.ilog.com/products/cplex>, Nov. 2004.
- [118]Intel Pentium 4 CPU @ 2GHz, 1Gb RAM.